



SAPIENZA
UNIVERSITÀ DI ROMA

Network Optimization

Exercises 3 — Flows

Worked exercises on the chapters of Volume 3, flows: each exercise is followed by its complete solution.

Fabio Furini

fabiofurini.github.io

Department of Computer, Control and Management Engineering — Sapienza University of Rome

Online version and downloadable material:
fabiofurini.github.io/network-optimization/
github.com/fabiofurini/network-optimization

Academic year 2026–2027

Contents

4	Network flow models	2
5	Maximum flow	12
6	Minimum cost flow	25
7	Transportation and assignment	40



Network flow models

Rules used in these exercises

The definitions and the results of the chapter «[Network flow models](#)», for a network $G = (N, A)$ with balances b_i , lower bounds l_{ij} , capacities u_{ij} and costs c_{ij} .

- (F1) **Model.** The arcs are numbered $a_1, a_2, \dots, a_m$ in lexicographic order; $\mathbf{B}$ has $+1$ in the tail and -1 in the head of every arc, and the MCF model is

$$\min\{\mathbf{c}'\mathbf{x} : \mathbf{B}\mathbf{x} = \mathbf{b}, \mathbf{l} \leq \mathbf{x} \leq \mathbf{u}\},$$

where row i of $\mathbf{B}\mathbf{x}$ is the outflow minus the inflow of node i .

- (F2) **Feasibility.** A network with a feasible flow has $\sum_{i \in N} b_i = 0$ and, for every $S \subseteq N$, $\sum_{i \in S} b_i \leq u(\delta^+(S)) - l(\delta^-(S))$.
- (F3) **Lower bounds.** With $\mathbf{x} = \mathbf{l} + \mathbf{x}'$: $\mathbf{b}' = \mathbf{b} - \mathbf{B}\mathbf{l}$, $\mathbf{u}' = \mathbf{u} - \mathbf{l}$, and $\mathbf{c}'\mathbf{x} = \mathbf{c}'\mathbf{x}' + \mathbf{c}'\mathbf{l}$.
- (F4) **Dummy node.** If $e = \sum_{i \in N} b_i > 0$, a dummy node d with $b_d = -e$ and arcs (i, d) from the supply nodes, with cost 0 and capacity $+\infty$; the flow on (i, d) is the unused supply of i .
- (F5) **Total unimodularity.** Every square submatrix of $\mathbf{B}$ has determinant 0 or ± 1 : a zero column gives 0, a column with one nonzero entry is expanded, columns with $+1$ and -1 give 0. With integer $\mathbf{b}, \mathbf{l}, \mathbf{u}$ every vertex of $\{\mathbf{B}\mathbf{x} = \mathbf{b}, \mathbf{l} \leq \mathbf{x} \leq \mathbf{u}\}$ is integer.

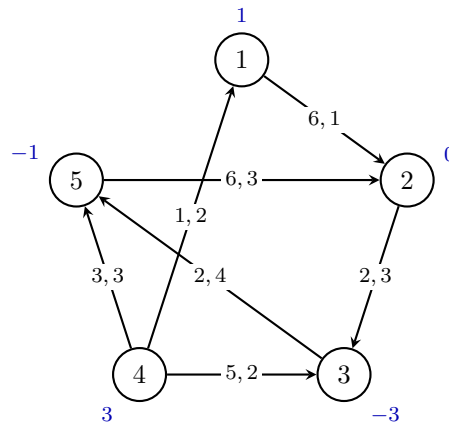
Exercise 4.1 — The minimum cost flow model

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 5\}$, the 7 arcs

$$A = \{(1, 2), (2, 3), (3, 5), (4, 1), (4, 3), (4, 5), (5, 2)\},$$

zero lower bounds and the balances $b_1 = 1, b_2 = 0, b_3 = -3, b_4 = 3, b_5 = -1$. Costs and capacities are given in the table and, as c_{ij}, u_{ij} , in the figure; the balances are next to the nodes.

arc	(1, 2)	(2, 3)	(3, 5)	(4, 1)	(4, 3)	(4, 5)	(5, 2)
c_{ij}	6	2	2	1	5	3	6
u_{ij}	1	3	4	2	2	3	3



Two flows are proposed:

arc	(1, 2)	(2, 3)	(3, 5)	(4, 1)	(4, 3)	(4, 5)	(5, 2)
$\bar{x}_{ij}$	1	3	2	0	2	1	2
$\hat{x}_{ij}$	1	3	3	0	2	1	2

Questions

1. Classify the nodes into supply, demand and transshipment nodes, and check that the network is balanced.
2. Number the arcs as in (F1), write the incidence matrix $\mathbf{B}$ and the MCF model explicitly.
3. Decide whether $\bar{\mathbf{x}}$ and $\hat{\mathbf{x}}$ are feasible flows, and compute the cost of the feasible one.

Solution

We use the rules (F1)–(F5) of the box at the beginning of these exercises.

1. By the sign of the balances: nodes 1, 4 are supply nodes; nodes 3, 5 are demand nodes; node 2 is a transshipment node. The total supply is 4, equal to the total demand, so $\sum_{i \in N} b_i = 0$.
2. The arcs in lexicographic order are $a_1 = (1, 2)$, $a_2 = (2, 3)$, $a_3 = (3, 5)$, $a_4 = (4, 1)$,

$a_5 = (4, 3)$, $a_6 = (4, 5)$, $a_7 = (5, 2)$. The incidence matrix is

	a_1	a_2	a_3	a_4	a_5	a_6	a_7
1	1	0	0	-1	0	0	0
2	-1	1	0	0	0	0	-1
3	0	-1	1	0	-1	0	0
4	0	0	0	1	1	1	0
5	0	0	-1	0	0	-1	1

and the MCF model has one equation per node (one row of $\mathbf{B}$):

$$\begin{aligned}
 \min \quad & 6x_{12} + 2x_{23} + 2x_{35} + x_{41} + 5x_{43} + 3x_{45} + 6x_{52} \\
 \text{subject to} \quad & x_{12} - x_{41} = 1 \\
 & -x_{12} + x_{23} - x_{52} = 0 \\
 & -x_{23} + x_{35} - x_{43} = -3 \\
 & x_{41} + x_{43} + x_{45} = 3 \\
 & -x_{35} - x_{45} + x_{52} = -1
 \end{aligned}$$

together with $0 \leq x_{12} \leq 1$, $0 \leq x_{23} \leq 3$, $0 \leq x_{35} \leq 4$, $0 \leq x_{41} \leq 2$, $0 \leq x_{43} \leq 2$, $0 \leq x_{45} \leq 3$, $0 \leq x_{52} \leq 3$.

3. We compute the outflow minus the inflow of every node, and compare it with $\mathbf{b}$:

$$\mathbf{B}\bar{\mathbf{x}} = (1, 0, -3, 3, -1)', \quad \mathbf{B}\hat{\mathbf{x}} = (1, 0, -2, 3, -2)', \quad \mathbf{b} = (1, 0, -3, 3, -1)'.$$

Both flows respect the bounds. The flow $\bar{\mathbf{x}}$ is feasible; $\hat{\mathbf{x}}$ is not, because it sends one more unit on $(3, 5)$ and conservation fails at nodes 3 and 5. The cost of $\bar{\mathbf{x}}$ is

$$6 \cdot 1 + 2 \cdot 3 + 2 \cdot 2 + 1 \cdot 0 + 5 \cdot 2 + 3 \cdot 1 + 6 \cdot 2 = 41,$$

and the arcs $(1, 2)$, $(2, 3)$, $(4, 3)$ are at capacity.

The MCF model has 7 variables and 5 equations with matrix $\mathbf{B}$; $\bar{\mathbf{x}}$ is feasible with cost 41, while $\hat{\mathbf{x}}$ violates conservation at nodes 3 and 5.

Exercise 4.2 — Feasibility conditions

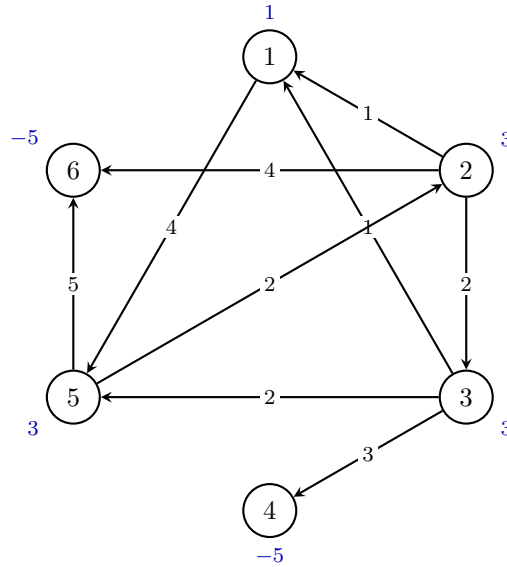
We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 6\}$, the 9 arcs

$$A = \{(1, 5), (2, 1), (2, 3), (2, 6), (3, 1), (3, 4), (3, 5), (5, 2), (5, 6)\},$$

zero lower bounds, the capacities u_{ij} of the table (next to the arcs in the figure) and the balances

$$b_1 = 1, \quad b_2 = 3, \quad b_3 = 3, \quad b_4 = -5, \quad b_5 = 3, \quad b_6 = -5.$$

arc	(1, 5)	(2, 1)	(2, 3)	(2, 6)	(3, 1)	(3, 4)	(3, 5)	(5, 2)	(5, 6)
u_{ij}	4	1	2	4	1	3	2	2	5



Questions

1. Is the network balanced?
2. For $S = \{1, 2, 3, 5, 6\}$ compute $\delta^+(S)$, $\delta^-(S)$ and check the cut condition of (F2). What can you conclude?
3. By how much must the capacity of the arc (3, 4) be raised for the cut condition to hold for S ? Check the condition for S with the new capacity.

Solution

We use the rules (F1)–(F5) of the box at the beginning of these exercises.

1. The balances add up to $1 + 3 + 3 + (-5) + 3 + (-5) = 0$: the network is balanced.
2. The arcs leaving S are $\delta^+(S) = \{(3, 4)\}$ and the arcs entering S are $\delta^-(S) = \emptyset$. Hence

$$\sum_{i \in S} b_i = 1 + 3 + 3 + 3 + (-5) = 5, \quad u(\delta^+(S)) = 3 = 3,$$

and the lower bounds of $\delta^-(S)$ are zero. Since $5 > 3$, the cut condition fails: the nodes of S have 5 units to send out and the arcs leaving S carry at most 3. The network is balanced but has no feasible flow.

3. The missing capacity is $5 - 3 = 2$, so u_{34} must go from 3 to 5. Then $u(\delta^+(S)) = 5 = 5$ and the condition holds for S with equality: every arc leaving S must be at capacity. With this capacity all the cut conditions hold and the network has a feasible flow.

$\sum_i b_i = 0$ is not enough: the set $S = \{1, 2, 3, 5, 6\}$ violates the cut condition by 2 units, and raising u_{34} to 5 repairs the network.

Exercise 4.3 — Removing the lower bounds

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 5\}$, the 7 arcs

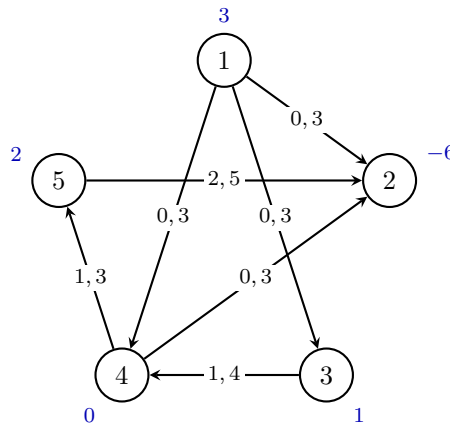
$$A = \{(1, 2), (1, 3), (1, 4), (3, 4), (4, 2), (4, 5), (5, 2)\},$$

the balances

$$b_1 = 3, \quad b_2 = -6, \quad b_3 = 1, \quad b_4 = 0, \quad b_5 = 2,$$

and the lower bounds, capacities and costs of the table (in the figure, l_{ij}, u_{ij} next to the arcs).

arc	(1, 2)	(1, 3)	(1, 4)	(3, 4)	(4, 2)	(4, 5)	(5, 2)
l_{ij}	0	0	0	1	0	1	2
u_{ij}	3	3	3	4	3	3	5
c_{ij}	5	4	6	2	6	4	5



Questions

1. Remove the lower bounds with the substitution $\mathbf{x} = \mathbf{l} + \mathbf{x}'$: compute $\mathbf{B}\mathbf{l}$, the new balances $\mathbf{b}'$ and the new capacities $\mathbf{u}'$.
2. Which constant must be added to the cost of a flow of the substituted network?
3. The flow $\mathbf{x}'$ below is feasible for the substituted network. Give the corresponding flow $\mathbf{x}$ of the original network and its cost.

arc	(1, 2)	(1, 3)	(1, 4)	(3, 4)	(4, 2)	(4, 5)	(5, 2)
x'_{ij}	1	1	1	1	2	0	1

Solution

We use the rules (F1)–(F5) of the box at the beginning of these exercises.

1. Each arc with $l_{ij} > 0$ takes l_{ij} units from its tail and gives them to its head: $(3, 4)$ with $l_{34} = 1$, $(4, 5)$ with $l_{45} = 1$, $(5, 2)$ with $l_{52} = 2$. Hence

$$\mathbf{Bl} = (0, -2, 1, 0, 1)', \quad \mathbf{b}' = \mathbf{b} - \mathbf{Bl} = (3, -4, 0, 0, 1)',$$

and the capacities decrease by the lower bounds:

arc	(1, 2)	(1, 3)	(1, 4)	(3, 4)	(4, 2)	(4, 5)	(5, 2)
u'_{ij}	3	3	3	3	3	2	3

The new balances still add up to 0.

2. The constant is $\mathbf{c}'\mathbf{l} = 2 \cdot 1 + 4 \cdot 1 + 5 \cdot 2 = 16$.
3. We add the lower bounds back, $x_{ij} = l_{ij} + x'_{ij}$:

arc	(1, 2)	(1, 3)	(1, 4)	(3, 4)	(4, 2)	(4, 5)	(5, 2)
x_{ij}	1	1	1	2	2	1	3

The cost of $\mathbf{x}'$ is 34, so the cost of $\mathbf{x}$ is $34 + 16 = 50$; computing $\mathbf{c}'\mathbf{x}$ directly gives the same value.

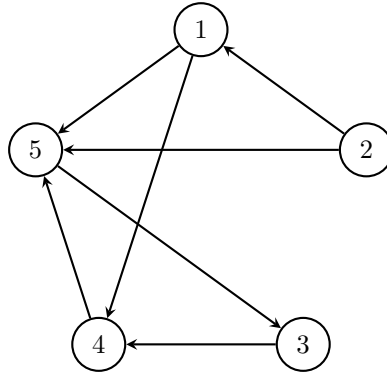
$\mathbf{b}' = (3, -4, 0, 0, 1)'$, $\mathbf{u}' = \mathbf{u} - \mathbf{l}$, and the costs of the two networks differ by $\mathbf{c}'\mathbf{l} = 16$.

Exercise 4.4 — Total unimodularity

We consider the directed graph $G = (N, A)$ with $N = \{1, 2, \dots, 5\}$ and the 7 arcs

$$A = \{(1, 4), (1, 5), (2, 1), (2, 5), (3, 4), (4, 5), (5, 3)\},$$

numbered $a_1, a_2, \dots, a_7$ in lexicographic order, and the undirected graph H with vertices $1, 2, 3, 4$ and edges $e_1 = \{1, 2\}$, $e_2 = \{1, 3\}$, $e_3 = \{2, 3\}$, $e_4 = \{3, 4\}$.



Questions

1. Write the node-arc incidence matrix $\mathbf{B}$ of G .
2. Compute the determinant of the submatrix of $\mathbf{B}$ with rows 1, 2, 5 and columns a_2, a_4, a_6 , following the cases of (F5).
3. Compute the determinant of the submatrix with rows 1, 2, 5 and columns a_2, a_3, a_4 . Which structure of the graph explains the value?
4. Write the vertex-edge incidence matrix $\mathbf{K}$ of H and show that it is not totally unimodular.
5. Let $\mathbf{b} \in \mathbb{Z}^5$ and $\mathbf{u} \in \mathbb{Z}_{\geq 0}^7$. Can the polyhedron $\{\mathbf{x} : \mathbf{B}\mathbf{x} = \mathbf{b}, \mathbf{0} \leq \mathbf{x} \leq \mathbf{u}\}$ have a fractional vertex?

Solution

We use the rules (F1)–(F5) of the box at the beginning of these exercises.

1. Each column has +1 in the tail and -1 in the head:

	a_1	a_2	a_3	a_4	a_5	a_6	a_7
1	1	1	-1	0	0	0	0
2	0	0	1	1	0	0	0
3	0	0	0	0	1	0	-1
4	-1	0	0	0	-1	1	0
5	0	-1	0	-1	0	-1	1

2. The submatrix is

	a_2	a_4	a_6
1	1	0	0
2	0	1	0
5	-1	-1	-1

- the column of a_6 has the single nonzero entry -1 in row 5: the expansion gives a factor -1 and deletes row 5 and the column;
- the column of a_2 has the single nonzero entry 1 in row 1: the expansion gives a factor 1 and deletes row 1 and the column;

The remaining 1×1 entry is 1; multiplying it by the factors, the determinant is -1 .

3. The submatrix is

	a_2	a_3	a_4
1	1	-1	0
2	0	1	1
5	-1	0	-1

Here every column has one $+1$ and one -1 : the rows add up to zero and the determinant is 0. The three arcs $(1, 5)$, $(2, 1)$, $(2, 5)$ have both endpoints among the nodes $1, 2, 5$: they form a cycle of the underlying undirected graph.

4. The rows are the vertices and the columns the edges e_1, e_2, e_3, e_4 :

	e_1	e_2	e_3	e_4
1	1	1	0	0
2	1	0	1	0
3	0	1	1	1
4	0	0	0	1

The submatrix of rows $1, 2, 3$ and columns e_1, e_2, e_3 (the triangle) has determinant $-2 \notin \{0, \pm 1\}$, so $\mathbf{K}$ is not totally unimodular. With this submatrix $\mathbf{T}$, the system $\mathbf{T}\mathbf{y} = \mathbf{1}$ has the unique solution $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$.

5. No: $\mathbf{B}$ is totally unimodular, and with integer $\mathbf{b}$ and $\mathbf{u}$ every vertex is integer by (F5). Computing all the 791 square submatrices of $\mathbf{B}$ confirms that their determinants are $-1, 0$ or 1 .

The two submatrices of $\mathbf{B}$ have determinants -1 and 0 ; the incidence matrix of the undirected graph H has a submatrix with determinant -2 and is not totally unimodular.

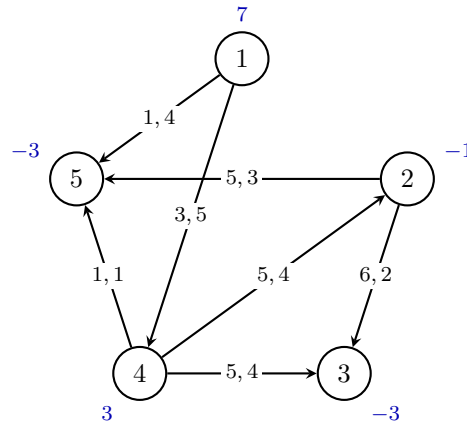
Exercise 4.5 — Unbalanced instances and the dummy node

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 5\}$, the 7 arcs

$$A = \{(1, 4), (1, 5), (2, 3), (2, 5), (4, 2), (4, 3), (4, 5)\},$$

zero lower bounds, the costs and capacities of the table (c_{ij}, u_{ij}) in the figure) and the balances $b_1 = 7, b_2 = -1, b_3 = -3, b_4 = 3, b_5 = -3$. Every demand must be met, and every supply node sends out at most its supply.

arc	(1, 4)	(1, 5)	(2, 3)	(2, 5)	(4, 2)	(4, 3)	(4, 5)
c_{ij}	3	1	6	5	5	5	1
u_{ij}	5	4	2	3	4	4	1



Questions

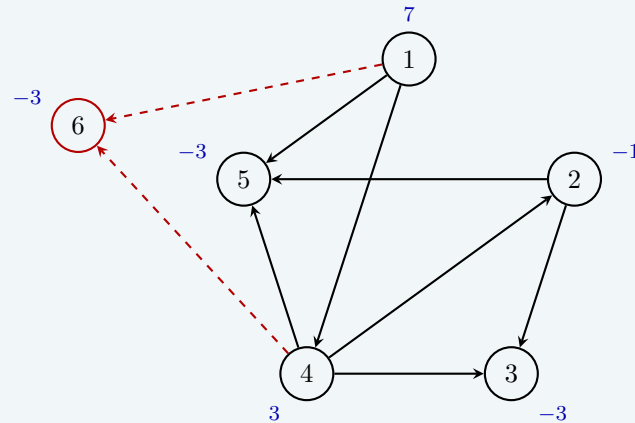
1. Compute the total supply, the total demand and the excess e . Is there a feasible flow with equality constraints?
2. Build the balanced network with a dummy node as in (F4): new node, balance, arcs, costs and capacities.
3. Write the conservation constraints of the dummy node and of the supply nodes in the balanced network.
4. An optimal flow of the balanced network, computed with an LP solver, is given below. How much supply does each supply node leave unused, and what is the optimal cost?

arc	(1, 4)	(1, 5)	(2, 3)	(2, 5)	(4, 2)	(4, 3)	(4, 5)	(1, 6)	(4, 6)
$\tilde{x}_{ij}$	1	3	0	0	1	3	0	3	0

Solution

We use the rules (F1)–(F5) of the box at the beginning of these exercises.

1. The total supply is $7 + 3 = 10$ and the total demand is 7, so $e = 10 - 7 = 3 > 0$. The balances do not add up to zero, and by (F2) no flow satisfies all the conservation equations.
2. We add node 6 with $b_6 = -3$ and the arcs (1, 6), (4, 6) with cost 0 and capacity $+\infty$ (dashed):



Now the balances add up to zero.

- The dummy node only receives flow, and each supply node has one more leaving arc:

$$\begin{aligned} -x_{16} - x_{46} &= -3, \\ x_{14} + x_{15} + x_{16} &= 7, \\ x_{42} + x_{43} + x_{45} + x_{46} - x_{14} &= 3. \end{aligned}$$

The flow on the dummy arc is the supply that node i does not send.

- The flows on the dummy arcs are $\tilde{x}_{16} = 3$, $\tilde{x}_{46} = 0$: node 1 sends 4 of its 7 units and leaves 3 unused; node 4 sends 3 of its 3 units and leaves 0 unused. The dummy arcs cost zero, so the optimal cost is that of the original arcs, 26.

The excess $e = 3$ goes to the dummy node 6 through zero-cost arcs; the optimal flow leaves 3 units at node 1 and 0 units at node 4, with cost 26.

Maximum flow

Rules used in these exercises

The definitions and the algorithms of the chapter «Maximum flow», for a directed graph $G = (N, A)$ without opposite arcs, with capacities u_{ij} , a source s and a sink t .

- (M1) **Flows and cuts.** A feasible flow has $0 \leq x_{ij} \leq u_{ij}$ and conservation at every node other than s and t ; its value v is the net outflow of s . For every s – t cut $(S, N \setminus S)$,

$$v = x(\delta^+(S)) - x(\delta^-(S)) \leq u(S, N \setminus S) = \sum_{(i,j) \in \delta^+(S)} u_{ij},$$

with equality if and only if the forward arcs of the cut are saturated and its backward arcs empty.

- (M2) **Residual network.** For $(i, j) \in A$: the forward arc (i, j) with $r_{ij} = u_{ij} - x_{ij}$ if $x_{ij} < u_{ij}$, the backward arc (j, i) with $r_{ji} = x_{ij}$ if $x_{ij} > 0$. Along an augmenting path P the bottleneck is $\delta = \min_{(i,j) \in P} r_{ij}$: x_{ij} grows by δ on the forward arcs of P and decreases by δ on the arcs of G whose backward arc is in P .
- (M3) **Search.** The search of $G(\mathbf{x})$ from s scans the residual arcs leaving each node by increasing index of the head, and stops as soon as t is labelled; the augmenting path is the path of the predecessors. Ford–Fulkerson uses the depth-first search ($LIST$ is a stack), Edmonds–Karp the breadth-first search ($LIST$ is a queue).
- (M4) **Stop.** If the search does not label t , the flow is maximum, and the set S of the labelled nodes gives a minimum cut, with $v = u(S, N \setminus S)$.
- (M5) **Edmonds–Karp.** The distances $d_{\mathbf{x}}(i)$ (fewest arcs from s in $G(\mathbf{x})$) never decrease; an arc of P with $r_{ij} = \delta$ is critical; at most nm augmentations.
- (M6) **Applications.** Matching: arcs (s, i) , (i, j) , (j, t) with capacity 1, the edges with flow 1 form a maximum matching. With all capacities 1, the maximum value is the maximum number of arc-disjoint paths from s to t , and the minimum number of arcs whose removal leaves no such path.

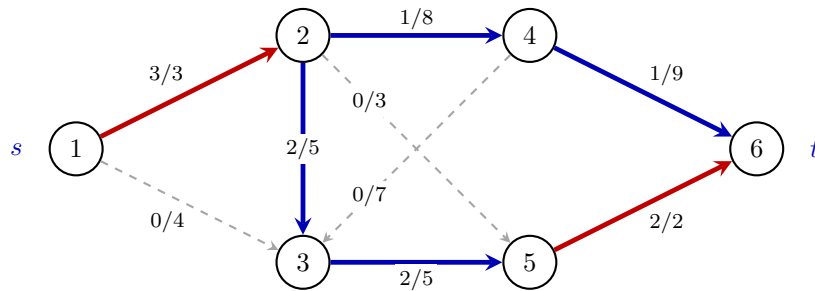
Exercise 5.1 — The residual network

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 6\}$, source $s = 1$, sink $t = 6$, the 9 arcs

$$A = \{(1, 2), (1, 3), (2, 3), (2, 4), (2, 5), (3, 5), (4, 3), (4, 6), (5, 6)\},$$

and the capacities u_{ij} of the table. A method that only increases the flows has stopped at the flow $\bar{x}$ of the table, drawn with $\bar{x}_{ij}/u_{ij}$ on the arcs.

arc	(1, 2)	(1, 3)	(2, 3)	(2, 4)	(2, 5)	(3, 5)	(4, 3)	(4, 6)	(5, 6)
u_{ij}	3	4	5	8	3	5	7	9	2
$\bar{x}_{ij}$	3	0	2	1	0	2	0	1	2


Questions

1. Check that $\bar{x}$ is a feasible flow and compute its value.
2. Draw the residual network $G(\bar{x})$, with the residual capacities.
3. Find the augmenting path given by the depth-first search of (M3), its bottleneck, and the flow after the augmentation.
4. Prove that the new flow is maximum.

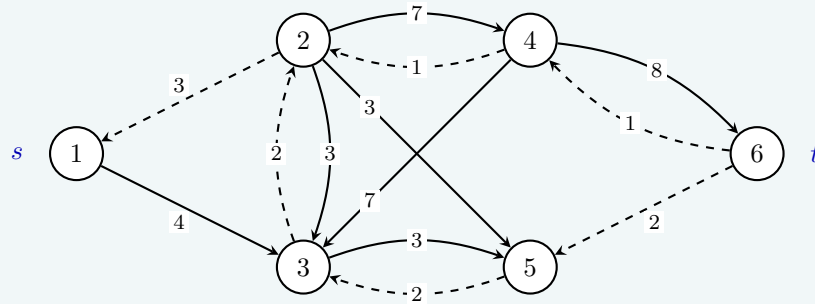
Solution

We use the rules (M1)–(M6) of the box at the beginning of these exercises.

1. All the flows are between 0 and the capacity. At the nodes $2, \dots, 5$ the inflow equals the outflow: node 2: $3 = 2 + 1 + 0$, node 3: $0 + 2 + 0 = 2$, node 4: $1 = 0 + 1$, node 5: $0 + 2 = 2$. The value is the net outflow of $s = 1$:

$$v = \bar{x}_{12} + \bar{x}_{13} = 3 + 0 = 3.$$

2. Every arc with $\bar{x}_{ij} < u_{ij}$ gives a forward arc with $r_{ij} = u_{ij} - \bar{x}_{ij}$, every arc with $\bar{x}_{ij} > 0$ a backward arc (j, i) with $r_{ji} = \bar{x}_{ij}$. The residual network has 13 arcs; the backward ones, dashed, are $(2, 1)$, $(3, 2)$, $(4, 2)$, $(5, 3)$, $(6, 4)$, $(6, 5)$.



3. The depth-first search labels 1, 3, 2, 4, 6, in this order, and finds

$$P = 1, 3, 2, 4, 6, \quad \delta = \min\{4, 2, 7, 8\} = 2,$$

through the backward arc (3, 2): the flow on (2, 3) decreases. The new flow is

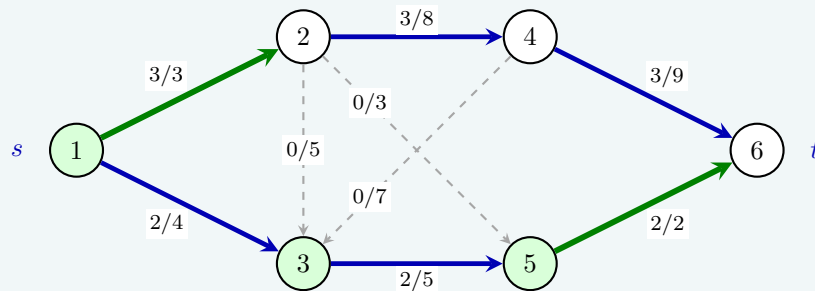
arc	(1, 2)	(1, 3)	(2, 3)	(2, 4)	(2, 5)	(3, 5)	(4, 3)	(4, 6)	(5, 6)
$\tilde{x}_{ij}$	3	2	0	3	0	2	0	3	2

with value $3 + 2 = 5$.

4. In $G(\tilde{\mathbf{x}})$ the search from 1 labels only the nodes of $S = \{1, 3, 5\}$, so there is no augmenting path. For $S = \{1, 3, 5\}$ the forward arcs (1, 2), (5, 6) are saturated, and the backward arcs (2, 3), (2, 5), (4, 3) are empty. Hence

$$u(S, N \setminus S) = 3 + 2 = 5 = v,$$

and by (M1) no flow has a larger value: $\tilde{\mathbf{x}}$ is maximum (the forward arcs of the cut are green, the nodes of S shaded).



The flow $\tilde{\mathbf{x}}$ of value 3 cannot be improved with forward arcs only; one augmentation through the backward arc (3, 2) reaches the maximum value 5, certified by the cut $S = \{1, 3, 5\}$.

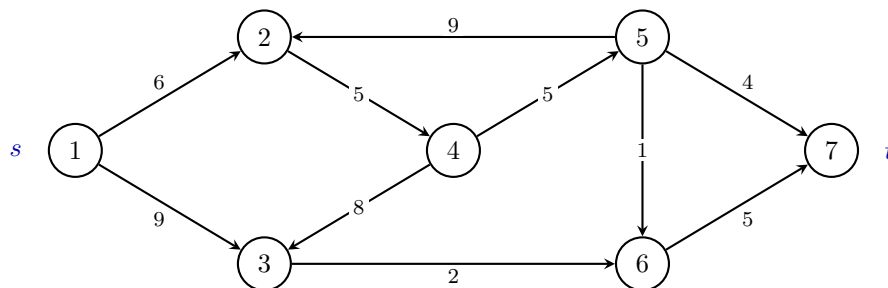
Exercise 5.2 — The Ford–Fulkerson algorithm

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 7\}$, source $s = 1$, sink $t = 7$, the 10 arcs

$$A = \{(1, 2), (1, 3), (2, 4), (3, 6), (4, 3), (4, 5), (5, 2), (5, 6), (5, 7), (6, 7)\},$$

and the capacities of the table, also next to the arcs in the figure.

arc	(1, 2)	(1, 3)	(2, 4)	(3, 6)	(4, 3)	(4, 5)	(5, 2)	(5, 6)	(5, 7)	(6, 7)
u_{ij}	6	9	5	2	8	5	9	1	4	5



Questions

1. Run the Ford–Fulkerson algorithm from the zero flow, with the depth-first search of (M3): for every iteration give the augmenting path, its residual capacities, δ and the new value.
2. Give the maximum flow and a minimum cut, and check that the value equals the capacity of the cut.
3. Which iteration uses a backward arc, and which flow of G does it decrease?

Solution

We use the rules (M1)–(M6) of the box at the beginning of these exercises.

1. Iteration 1

The search labels the nodes 1, 2, 4, 3, 6, 7, in this order. The augmenting path is $P = 1, 2, 4, 3, 6, 7$; its residual capacities are 6, 5, 8, 2, 5, so $\delta = 2$. New flows: $x_{12} = 0 + 2 = 2$, $x_{24} = 0 + 2 = 2$, $x_{43} = 0 + 2 = 2$, $x_{36} = 0 + 2 = 2$, $x_{67} = 0 + 2 = 2$; the value becomes $v = 2$.

Iteration 2

The search labels the nodes 1, 2, 4, 3, 5, 6, 7, in this order. The augmenting path is $P = 1, 2, 4, 5, 6, 7$; its residual capacities are 4, 3, 5, 1, 3, so $\delta = 1$. New flows: $x_{12} = 2 + 1 = 3$, $x_{24} = 2 + 1 = 3$, $x_{45} = 0 + 1 = 1$, $x_{56} = 0 + 1 = 1$, $x_{67} = 2 + 1 = 3$; the value becomes $v = 3$.

Iteration 3

The search labels the nodes 1, 2, 4, 3, 5, 7, in this order. The augmenting path is $P = 1, 2, 4, 5, 7$; its residual capacities are 3, 2, 4, 4, so $\delta = 2$. New flows: $x_{12} = 3 + 2 = 5$, $x_{24} = 3 + 2 = 5$, $x_{45} = 1 + 2 = 3$, $x_{57} = 0 + 2 = 2$; the value becomes $v = 5$.

Iteration 4

The search labels the nodes 1, 2, 3, 4, 5, 7, in this order. The augmenting path is $P = 1, 3, 4, 5, 7$; its residual capacities are 9, 2, 2, 2, so $\delta = 2$. It uses the backward arc $(3, 4)$. New flows: $x_{13} = 0 + 2 = 2$, $x_{43} = 2 - 2 = 0$, $x_{45} = 3 + 2 = 5$, $x_{57} = 2 + 2 = 4$; the value becomes $v = 7$.

Iteration 5

The search labels only the nodes of $S = \{1, 2, 3\}$: t is not labelled, and the algorithm stops with $v = 7$.

In short:

iteration	augmenting path	r_{ij} on P	backward arcs	δ	v
1	1, 2, 4, 3, 6, 7	6, 5, 8, 2, 5	—	2	2
2	1, 2, 4, 5, 6, 7	4, 3, 5, 1, 3	—	1	3
3	1, 2, 4, 5, 7	3, 2, 4, 4	—	2	5
4	1, 3, 4, 5, 7	9, 2, 2, 2	(3, 4)	2	7

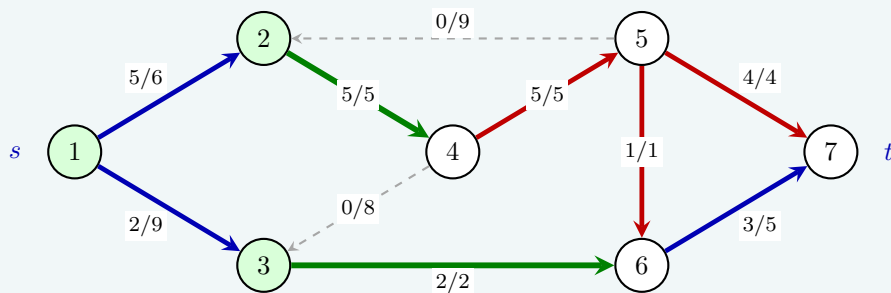
2. The maximum flow is

arc	(1, 2)	(1, 3)	(2, 4)	(3, 6)	(4, 3)	(4, 5)	(5, 2)	(5, 6)	(5, 7)	(6, 7)
$\tilde{x}_{ij}$	5	2	5	2	0	5	0	1	4	3
u_{ij}	6	9	5	2	8	5	9	1	4	5

with value $\tilde{v} = 7$. The last search labels $S = \{1, 2, 3\}$. For $S = \{1, 2, 3\}$ the forward arcs $(2, 4)$, $(3, 6)$ are saturated, and the backward arcs $(4, 3)$, $(5, 2)$ are empty. Hence

$$u(S, N \setminus S) = 5 + 2 = 7 = v,$$

so the flow is maximum and the cut is minimum. In the figure the forward arcs of the cut are green and the nodes of S shaded.



Uniqueness. The minimum cuts, of capacity 7, are $S = \{1, 2, 3\}$, $S = \{1, 2, 3, 4\}$, $S = \{1, 2, 3, 4, 5\}$. By (M1) every maximum flow saturates their forward arcs and empties their backward arcs: $x_{24} = 5$, $x_{36} = 2$, $x_{43} = 0$, $x_{45} = 5$, $x_{52} = 0$, $x_{56} = 1$, $x_{57} = 4$. Conservation then fixes the other arcs: at node 2, $x_{12} = 5$; at node 3, $x_{13} = 2$; at node 6, $x_{67} = 3$. Hence $\tilde{x}$ is the only maximum flow.

- Iteration 4 uses the backward arc $(3, 4)$: the flow on $(4, 3)$ decreases by $\delta = 2$, undoing part of an earlier choice of the depth-first search.

Ford–Fulkerson makes 4 augmentations and reaches $\tilde{v} = 7$; the last search gives the minimum cut $S = \{1, 2, 3\}$ of capacity 7.

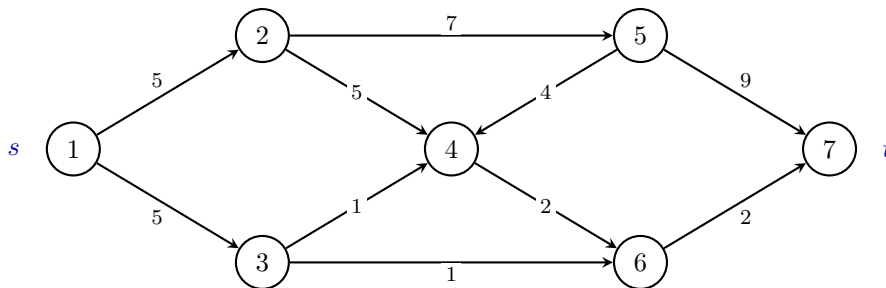
Exercise 5.3 — The Edmonds–Karp algorithm

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 7\}$, source $s = 1$, sink $t = 7$, the 10 arcs

$$A = \{(1, 2), (1, 3), (2, 4), (2, 5), (3, 4), (3, 6), (4, 6), (5, 4), (5, 7), (6, 7)\},$$

and the capacities of the table, also next to the arcs in the figure.

arc	(1, 2)	(1, 3)	(2, 4)	(2, 5)	(3, 4)	(3, 6)	(4, 6)	(5, 4)	(5, 7)	(6, 7)
u_{ij}	5	5	5	7	1	1	2	4	9	2



Questions

- Run the Edmonds–Karp algorithm from the zero flow: for every iteration give the augmenting path, its number of arcs, its residual capacities, δ and the new value.
- Give the distances $d_x(i)$ of all the nodes in the residual network before each search, check that they never decrease, and list the critical arcs of every augmentation.
- Give the maximum flow and a minimum cut. On this network the Ford–Fulkerson algorithm with the depth-first search makes 4 augmentations: compare.

Solution

We use the rules (M1)–(M6) of the box at the beginning of these exercises.

1. Iteration 1

The search labels the nodes 1, 2, 3, 4, 5, 6, 7, in this order. The augmenting path is $P = 1, 2, 5, 7$, with 3 arcs; its residual capacities are 5, 7, 9, so $\delta = 5$. New flows: $x_{12} = 0 + 5 = 5$, $x_{25} = 0 + 5 = 5$, $x_{57} = 0 + 5 = 5$; the value becomes $v = 5$.

Iteration 2

The search labels the nodes 1, 3, 4, 6, 7, in this order. The augmenting path is $P = 1, 3, 6, 7$, with 3 arcs; its residual capacities are 5, 1, 2, so $\delta = 1$. New flows: $x_{13} = 0 + 1 = 1$, $x_{36} = 0 + 1 = 1$, $x_{67} = 0 + 1 = 1$; the value becomes $v = 6$.

Iteration 3

The search labels the nodes 1, 3, 4, 6, 7, in this order. The augmenting path is $P = 1, 3, 4, 6, 7$, with 4 arcs; its residual capacities are 4, 1, 2, 1, so $\delta = 1$. New flows: $x_{13} = 1 + 1 = 2$, $x_{34} = 0 + 1 = 1$, $x_{46} = 0 + 1 = 1$, $x_{67} = 1 + 1 = 2$; the value becomes $v = 7$.

Iteration 4

The search labels only the nodes of $S = \{1, 3\}$: t is not labelled, and the algorithm stops with $v = 7$.

2. The distances are computed by a full breadth-first search of the residual network:

before the search	$d(1)$	$d(2)$	$d(3)$	$d(4)$	$d(5)$	$d(6)$	$d(7)$
iteration 1	0	1	1	2	2	2	3
iteration 2	0	5	1	2	4	2	3
iteration 3	0	6	1	2	5	3	4
stop	0	∞	1	∞	∞	∞	∞

Every column is non-decreasing, as (M5) states. The critical arcs, with $r_{ij} = \delta$, are: iteration 1: (1, 2); iteration 2: (3, 6); iteration 3: (3, 4), (6, 7).

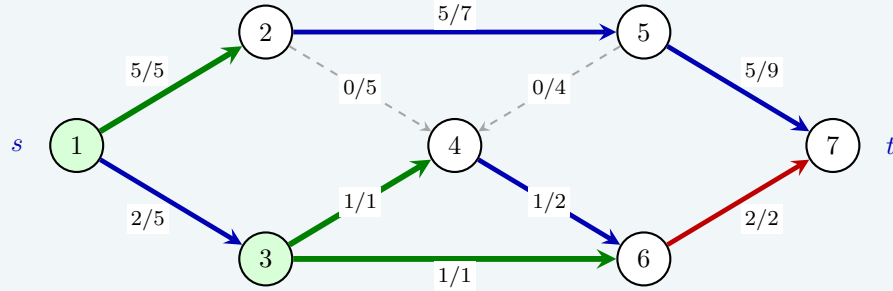
3. The maximum flow is

arc	(1, 2)	(1, 3)	(2, 4)	(2, 5)	(3, 4)	(3, 6)	(4, 6)	(5, 4)	(5, 7)	(6, 7)
$\tilde{x}_{ij}$	5	2	0	5	1	1	1	0	5	2
u_{ij}	5	5	5	7	1	1	2	4	9	2

with value $\tilde{v} = 7$. The last search labels $S = \{1, 3\}$. For $S = \{1, 3\}$ the forward arcs (1, 2), (3, 4), (3, 6) are saturated, and there are no backward arcs. Hence

$$u(S, N \setminus S) = 5 + 1 + 1 = 7 = v,$$

so the cut is minimum (forward arcs in green, nodes of S shaded).



Edmonds–Karp needs 3 augmentations, against the 4 of Ford–Fulkerson; both are far below the bound $nm = 70$ of (M5).

Uniqueness. The minimum cuts, of capacity 7, are $S = \{1, 3\}$, $S = \{1, 3, 4, 6\}$. By (M1) every maximum flow saturates their forward arcs and empties their backward arcs: $x_{12} = 5$, $x_{24} = 0$, $x_{34} = 1$, $x_{36} = 1$, $x_{54} = 0$, $x_{67} = 2$. Conservation then fixes the other arcs: at node 1, $x_{13} = 2$; at node 2, $x_{25} = 5$; at node 4, $x_{46} = 1$; at node 5, $x_{57} = 5$. Hence $\tilde{x}$ is the only maximum flow.

Edmonds–Karp reaches $\tilde{v} = 7$ in 3 augmentations along paths with 3,3,4 arcs; the distances never decrease, and the cut $S = \{1, 3\}$ is minimum.

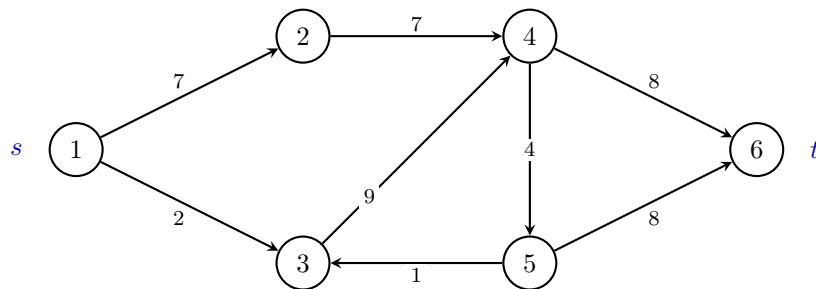
Exercise 5.4 — Minimum cuts

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 6\}$, source $s = 1$, sink $t = 6$, the 8 arcs

$$A = \{(1, 2), (1, 3), (2, 4), (3, 4), (4, 5), (4, 6), (5, 3), (5, 6)\},$$

and the integer capacities of the table, also next to the arcs in the figure.

arc	(1, 2)	(1, 3)	(2, 4)	(3, 4)	(4, 5)	(4, 6)	(5, 3)	(5, 6)
u_{ij}	7	2	7	9	4	8	1	8



Questions

1. List the s – t cuts with their capacities, and find all the minimum cuts.
2. Check that the flow x below is maximum.

arc	(1, 2)	(1, 3)	(2, 4)	(3, 4)	(4, 5)	(4, 6)	(5, 3)	(5, 6)
x_{ij}	7	2	7	2	1	8	0	1

3. Prove that, with integer capacities, raising u_{ij} by one unit raises the maximum value if and only if (i, j) is a forward arc of every minimum cut.
4. For which arcs of this network does raising the capacity by one unit raise the maximum value?

Solution

We use the rules (M1)–(M6) of the box at the beginning of these exercises.

1. There are $2^4 = 16$ cuts; next to each set S , its capacity:

S	u	S	u	S	u	S	u
$\{1\}$	9	$\{1, 2\}$	9	$\{1, 3\}$	16	$\{1, 4\}$	21
$\{1, 5\}$	18	$\{1, 2, 3\}$	16	$\{1, 2, 4\}$	14	$\{1, 2, 5\}$	18
$\{1, 3, 4\}$	19	$\{1, 3, 5\}$	24	$\{1, 4, 5\}$	26	$\{1, 2, 3, 4\}$	12
$\{1, 2, 3, 5\}$	24	$\{1, 2, 4, 5\}$	19	$\{1, 3, 4, 5\}$	23	$\{1, 2, 3, 4, 5\}$	16

The minimum capacity is 9, attained by the 2 cuts $S = \{1\}$, $S = \{1, 2\}$.

2. The flow is feasible: every arc carries a flow between 0 and its capacity, and at every node other than 1 and 6 the inflow equals the outflow. Its value is 9, equal to the minimum cut capacity: by (M1) no flow is better. For example, for $S = \{1\}$ the forward arcs $(1, 2)$, $(1, 3)$ are saturated and the backward arcs do not exist.
3. Let $\tilde{v}$ be the maximum value and $\mathbf{u}'$ the capacities with $u'_{ij} = u_{ij} + 1$. The capacity of a cut grows by 1 if (i, j) is one of its forward arcs, and does not change otherwise.
 - If (i, j) is a forward arc of every minimum cut, every minimum cut gets capacity $\tilde{v} + 1$. Every other cut had an integer capacity larger than $\tilde{v}$, so at least $\tilde{v} + 1$, and it does not decrease. The new minimum cut capacity is $\tilde{v} + 1$, and by the max-flow min-cut theorem so is the new maximum value.
 - If a minimum cut $(S, N \setminus S)$ does not have (i, j) as a forward arc, its capacity stays $\tilde{v}$, and every flow for $\mathbf{u}'$ has value at most $\tilde{v}$ by (M1). The old maximum flow is still feasible, so the maximum value stays $\tilde{v}$.

With fractional capacities the first part fails: a cut that is not minimum may be less than one unit above $\tilde{v}$.

4. By question 3, we look for the arcs that are forward arcs of all the minimum cuts:

arc	forward arc of	maximum value with $u_{ij} + 1$
(1, 2)	$S = \{1\}$	9
(1, 3)	every minimum cut	10
(2, 4)	$S = \{1, 2\}$	9
(3, 4)	no minimum cut	9
(4, 5)	no minimum cut	9
(4, 6)	no minimum cut	9
(5, 3)	no minimum cut	9
(5, 6)	no minimum cut	9

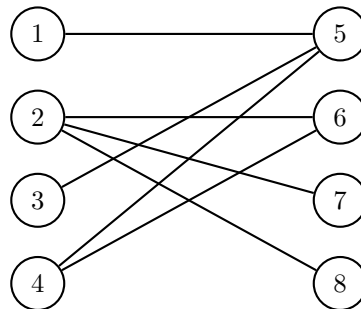
The maximum value grows only for the arc (1, 3), from 9 to 10. The arcs (1, 2), (2, 4) belong to some minimum cuts but not to all, and raising them changes nothing.

The maximum value is 9, with 2 minimum cuts; only the arc (1, 3) belongs to all of them, and only raising its capacity raises the maximum value.

Exercise 5.5 — Maximum matching

Four workers $V_1 = \{1, 2, 3, 4\}$ and four jobs $V_2 = \{5, 6, 7, 8\}$ form the bipartite graph $H = (V_1 \cup V_2, E)$, with an edge $\{i, j\}$ when worker i can do job j :

$$E = \{\{1, 5\}, \{2, 6\}, \{2, 7\}, \{2, 8\}, \{3, 5\}, \{4, 5\}, \{4, 6\}\}.$$



Questions

1. Build the network of H (M6), with the nodes in the order $s, 1, 2, \dots, 8, t$.
2. Run the Edmonds–Karp algorithm on it from the zero flow.
3. Give a maximum matching and the minimum cut found at the stop.
4. Let S be the set labelled by the last search, and $C = (V_1 \setminus S) \cup (V_2 \cap S)$. Check that every edge has an endpoint in C , and deduce again that no matching has more edges than the one found.

Solution

We use the rules (M1)–(M6) of the box at the beginning of these exercises.

1. The network has the arcs (s, i) for $i \in V_1$, the arcs (i, j) for the 7 edges, and the arcs (j, t) for $j \in V_2$, all with capacity 1: 15 arcs in all.

2. Iteration 1

The search labels the nodes $s, 1, 2, 3, 4, 5, 6, 7, 8, t$, in this order. The augmenting path is $P = s, 1, 5, t$, with 3 arcs; its residual capacities are 1, 1, 1, so $\delta = 1$. New flows: $x_{s1} = 0 + 1 = 1$, $x_{15} = 0 + 1 = 1$, $x_{5t} = 0 + 1 = 1$; the value becomes $v = 1$.

Iteration 2

The search labels the nodes $s, 2, 3, 4, 6, 7, 8, 5, t$, in this order. The augmenting path is $P = s, 2, 6, t$, with 3 arcs; its residual capacities are 1, 1, 1, so $\delta = 1$. New flows: $x_{s2} = 0 + 1 = 1$, $x_{26} = 0 + 1 = 1$, $x_{6t} = 0 + 1 = 1$; the value becomes $v = 2$.

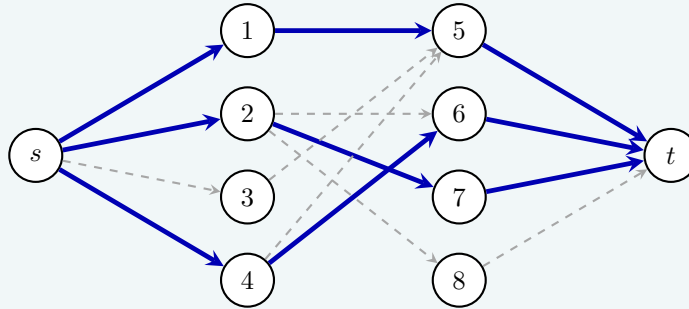
Iteration 3

The search labels the nodes $s, 3, 4, 5, 6, 1, 2, 7, 8, t$, in this order. The augmenting path is $P = s, 4, 6, 2, 7, t$, with 5 arcs; its residual capacities are 1, 1, 1, 1, 1, so $\delta = 1$. It uses the backward arc $(6, 2)$. New flows: $x_{s4} = 0 + 1 = 1$, $x_{46} = 0 + 1 = 1$, $x_{26} = 1 - 1 = 0$, $x_{27} = 0 + 1 = 1$, $x_{7t} = 0 + 1 = 1$; the value becomes $v = 3$.

Iteration 4

The search labels only the nodes of $S = \{s, 1, 3, 5\}$: t is not labelled, and the algorithm stops with $v = 3$.

3. The edges with flow 1 give the matching $M = \{\{1, 5\}, \{2, 7\}, \{4, 6\}\}$, with 3 edges; it is not the only maximum matching (there are 4). Iteration 3 reassigns jobs through the backward arc $(6, 2)$. The last search labels $S = \{s, 1, 3, 5\}$; the forward arcs of the cut are $(s, 2)$, $(s, 4)$, $(5, t)$, with capacity $3 = |M|$, so the matching is maximum.



4. Here $C = \{2, 4, 5\}$, with 3 nodes. Every edge $\{i, j\}$ has $i \in C$ or $j \in C$: $\{1, 5\}$ by 5, $\{2, 6\}$ by 2, $\{2, 7\}$ by 2, $\{2, 8\}$ by 2, $\{3, 5\}$ by 5, $\{4, 5\}$ by 4, $\{4, 6\}$ by 4. The edges of a matching have no common endpoint, so each of them uses a different node of C : no matching has more than $|C| = 3$ edges.

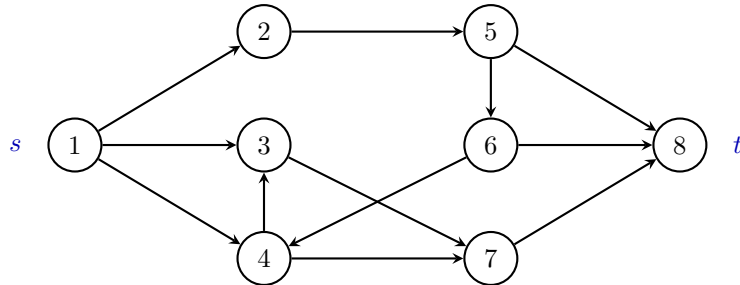
The maximum matching has 3 edges, one job stays free; the cut of the last search and the 3 nodes of C certify that no larger matching exists.

Exercise 5.6 — Arc-disjoint paths

We consider the directed graph $G = (N, A)$ with $N = \{1, 2, \dots, 8\}$ and the 12 arcs

$$A = \{(1, 2), (1, 3), (1, 4), (2, 5), (3, 7), (4, 3), (4, 7), (5, 6), (5, 8), (6, 4), (6, 8), (7, 8)\}.$$

We look for paths from $s = 1$ to $t = 8$ with no common arc.



Questions

1. Give every arc the capacity 1 and run the Edmonds–Karp algorithm from the zero flow.
2. Decompose the maximum flow into arc-disjoint paths from 1 to 8.
3. Find a minimum set of arcs whose removal leaves no path from 1 to 8. Is the set of the arcs leaving node 1 such a minimum set?

Solution

We use the rules (M1)–(M6) of the box at the beginning of these exercises.

1. Iteration 1

The search labels the nodes 1, 2, 3, 4, 5, 7, 6, 8, in this order. The augmenting path is $P = 1, 2, 5, 8$, with 3 arcs; its residual capacities are 1, 1, 1, so $\delta = 1$. New flows: $x_{12} = 0 + 1 = 1$, $x_{25} = 0 + 1 = 1$, $x_{58} = 0 + 1 = 1$; the value becomes $v = 1$.

Iteration 2

The search labels the nodes 1, 3, 4, 7, 8, in this order. The augmenting path is $P = 1, 3, 7, 8$, with 3 arcs; its residual capacities are 1, 1, 1, so $\delta = 1$. New flows: $x_{13} = 0 + 1 = 1$, $x_{37} = 0 + 1 = 1$, $x_{78} = 0 + 1 = 1$; the value becomes $v = 2$.

Iteration 3

The search labels only the nodes of $S = \{1, 3, 4, 7\}$: t is not labelled, and the algorithm stops with $v = 2$.

2. The arcs with flow 1 are $(1, 2)$, $(1, 3)$, $(2, 5)$, $(3, 7)$, $(5, 8)$, $(7, 8)$. Following them from 1, each arc once, gives the paths

$$1, 2, 5, 8, \quad 1, 3, 7, 8,$$

with no common arc. By (M6) no family of arc-disjoint paths has more than 2 paths.

3. The last search labels $S = \{1, 3, 4, 7\}$, and the forward arcs of the cut are $(1, 2)$, $(7, 8)$: removing these 2 arcs leaves no path from 1 to 8, since every such path leaves S through one of them. Fewer arcs are not enough, because each of the 2 arc-disjoint paths needs its own removed arc. The 3 arcs leaving node 1 also disconnect 1 from 8, but they are not a minimum set: the cut $S = \{1\}$ has capacity $3 > 2$. The minimum cuts are $S = \{1, 3, 4, 7\}$, $S = \{1, 2, 3, 4, 7\}$.

At most 2 arc-disjoint paths go from 1 to 8, and the 2 arcs of the cut $S = \{1, 3, 4, 7\}$ disconnect them.

Minimum cost flow

Rules used in these exercises

The definitions and the algorithms of the chapter «Minimum cost flow», for a balanced network $G = (N, A)$ with zero lower bounds, capacities u_{ij} and costs c_{ij} ; the arcs are in lexicographic order.

(M1) **Residual network.** For $(i, j) \in A$: the forward arc (i, j) with $r_{ij} = u_{ij} - x_{ij}$ and cost c_{ij} if $x_{ij} < u_{ij}$; the backward arc (j, i) with $r_{ji} = x_{ij}$ and cost $-c_{ij}$ if $x_{ij} > 0$. A cycle W of $G(\mathbf{x})$ has cost $c(W)$, the sum of the costs, and residual capacity $r(W)$, the minimum of the r ; augmenting by $\theta \leq r(W)$ changes the cost by $\theta c(W)$.

(M2) **Optimality.** A feasible flow is optimal if and only if $G(\mathbf{x})$ has no negative cycle, if and only if some π gives $c_{ij}^x - \pi_i + \pi_j \geq 0$ on every arc of $G(\mathbf{x})$, that is $\bar{c}_{ij} = c_{ij} - \pi_i + \pi_j > 0 \Rightarrow x_{ij} = 0$ and $\bar{c}_{ij} < 0 \Rightarrow x_{ij} = u_{ij}$.

(M3) **Bellman–Ford on $G(\mathbf{x})$.** $d^0(j) = 0$ for every j and

$$d^k(j) = \min \left\{ d^{k-1}(j), \min_{(i,j) \in A(\mathbf{x})} (d^{k-1}(i) + c_{ij}^x) \right\};$$

ties go to the smallest tail, and a predecessor changes only on a strict decrease. If $d^k = d^{k-1}$, stop: no negative cycle, and $\pi_i = -d^k(i)$, shifted so that $\pi_n = 0$. If $d^n \neq d^{n-1}$: j is the smallest node with $d^n(j) < d^{n-1}(j)$, the walk is read backwards from row n to row 1, and the cycle is the piece between the two occurrences of the first node met twice, reading the walk from its first node.

(M4) **Cycle canceling.** Augment along the cycle of (M3) by $\theta = r(W)$, until (M3) finds no negative cycle.

(M5) **First feasible flow and decomposition.** Arcs (s, i) with capacity b_i for $b_i > 0$, arcs (i, t) with capacity $-b_i$ for $b_i < 0$; the Edmonds–Karp algorithm: breadth-first search from s , the arcs leaving a node scanned by increasing head (s first, t last). A circulation is decomposed starting from its first arc with $y > 0$ in lexicographic order and leaving every node along its arc with $y > 0$ of smallest head.

(M6) **Successive shortest paths.** Start with $\mathbf{x} = \mathbf{0}$, $\pi = \mathbf{0}$, $\mathbf{e} = \mathbf{b}$; s is the node of smallest index with $e_s > 0$; Dijkstra from s with lengths $c_{ij}^x - \pi_i + \pi_j$ (smallest index on ties); t is the node with $e_t < 0$ of smallest distance (smallest index on ties); $\pi_i := \pi_i - \min\{d(i), d(t)\}$ and $\delta = \min\{e_s, -e_t, r(P)\}$.

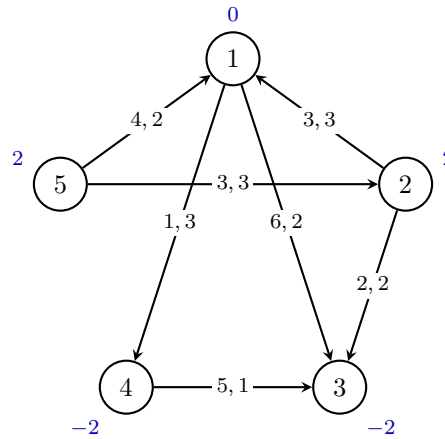
Exercise 6.1 — The residual network

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 5\}$, the 7 arcs

$$A = \{(1, 3), (1, 4), (2, 1), (2, 3), (4, 3), (5, 1), (5, 2)\},$$

zero lower bounds and the balances $b_1 = 0$, $b_2 = 2$, $b_3 = -2$, $b_4 = -2$, $b_5 = 2$. Costs and capacities are given in the table and, as c_{ij}, u_{ij} , in the figure; the balances are next to the nodes. The table also gives a feasible flow $\bar{x}$.

arc (i, j)	$(1, 3)$	$(1, 4)$	$(2, 1)$	$(2, 3)$	$(4, 3)$	$(5, 1)$	$(5, 2)$
c_{ij}	6	1	3	2	5	4	3
u_{ij}	2	3	3	2	1	2	3
$\bar{x}_{ij}$	0	3	2	1	1	1	1



The sequences $W_1 = 1, 2, 3, 4, 1$ and $W_2 = 1, 3, 2, 1$ are cycles of the residual network $G(\bar{x})$.

Questions

1. Build $G(\bar{x})$: for every arc of A give its forward and backward arcs, with residual capacity and cost.
2. Compute the cost $c(W)$ and the residual capacity $r(W)$ of W_1 and of W_2 . Which of the two cycles can lower the cost of $\bar{x}$?
3. Augment $\bar{x}$ along that cycle by $\theta = r(W)$. Give the new flow, check that it is feasible, and compute its cost in two ways.

Solution

We use the rules (M1)–(M6) of the box at the beginning of these exercises.

1. By (M1) the arc (i, j) gives the forward arc if $\bar{x}_{ij} < u_{ij}$ and the backward arc (j, i) if $\bar{x}_{ij} > 0$; each entry is r, c :

arc (i, j)	(1, 3)	(1, 4)	(2, 1)	(2, 3)	(4, 3)	(5, 1)	(5, 2)
$\bar{x}_{ij}$	0	3	2	1	1	1	1
forward $(i, j): r, c$	2, 6	–	1, 3	1, 2	–	1, 4	2, 3
backward $(j, i): r, c$	–	3, –1	2, –3	1, –2	1, –5	1, –4	1, –3

$G(\bar{\mathbf{x}})$ has 11 arcs.

2. The cycle $W_1 = 1, 2, 3, 4, 1$ uses the arcs (1, 2) backward of (2, 1), (2, 3) forward, (3, 4) backward of (4, 3), (4, 1) backward of (1, 4):

$$c(W) = -3 + 2 - 5 - 1 = -7, \quad r(W) = \min\{2, 1, 1, 3\} = 1.$$

The cycle $W_2 = 1, 3, 2, 1$ uses the arcs (1, 3) forward, (3, 2) backward of (2, 3), (2, 1) forward:

$$c(W) = 6 - 2 + 3 = 7, \quad r(W) = \min\{2, 1, 1\} = 1.$$

Only the negative cycle W_1 lowers the cost: augmenting along it by θ changes the cost by $\theta c(W) < 0$, while W_2 would raise it.

3. With $\theta = 1$, x_{ij} grows by 1 on the forward arcs of the cycle and decreases by 1 on the arcs whose backward arc is in the cycle:

arc (i, j)	(1, 3)	(1, 4)	(2, 1)	(2, 3)	(4, 3)	(5, 1)	(5, 2)
$\bar{x}_{ij}$	0	3	2	1	1	1	1
x'_{ij}	0	2	1	2	0	1	1

The bounds hold, and every node keeps its outflow minus inflow, equal to b_i : $\mathbf{x}'$ is feasible. Directly, $\mathbf{c}'\mathbf{x}' = 6 \cdot 0 + 1 \cdot 2 + 3 \cdot 1 + 2 \cdot 2 + 5 \cdot 0 + 4 \cdot 1 + 3 \cdot 1 = 16$; with (M1), $\mathbf{c}'\bar{\mathbf{x}} + \theta c(W) = 23 + 1 \cdot (-7) = 16$. The arcs that change are (1, 4), (2, 1), (2, 3), (4, 3).

$G(\bar{\mathbf{x}})$ has 11 arcs; the cycle 1, 2, 3, 4, 1 costs -7 with residual capacity 1, and augmenting along it lowers the cost from 23 to 16: $\bar{\mathbf{x}}$ is not optimal.

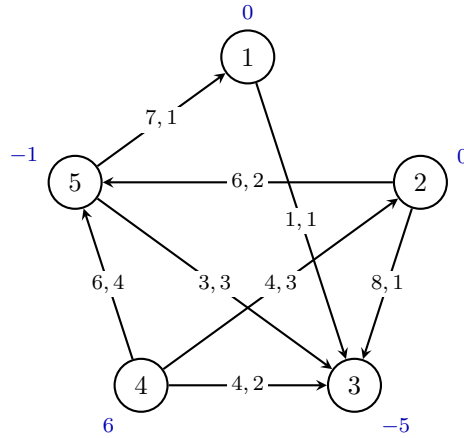
Exercise 6.2 — Optimality conditions

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 5\}$, the 8 arcs

$$A = \{(1, 3), (2, 3), (2, 5), (4, 2), (4, 3), (4, 5), (5, 1), (5, 3)\},$$

zero lower bounds and the balances $b_1 = 0$, $b_2 = 0$, $b_3 = -5$, $b_4 = 6$, $b_5 = -1$. Costs and capacities are given in the table and, as c_{ij}, u_{ij} , in the figure; the balances are next to the nodes. The table also gives a feasible flow $\tilde{\mathbf{x}}$, claimed to be optimal.

arc (i, j)	(1, 3)	(2, 3)	(2, 5)	(4, 2)	(4, 3)	(4, 5)	(5, 1)	(5, 3)
c_{ij}	1	8	6	4	4	6	7	3
u_{ij}	1	1	2	3	2	4	1	3
$\tilde{x}_{ij}$	0	0	0	0	2	4	0	3



Questions

1. Build the residual network $G(\tilde{\mathbf{x}})$.
2. Run Bellman–Ford on $G(\tilde{\mathbf{x}})$ from $d^0 = \mathbf{0}$, as in (M3). What do you conclude?
3. Give the potentials π of (M3), compute the reduced costs $\bar{c}_{ij}$ of the arcs of A , and check the conditions of (M2) arc by arc.

Solution

We use the rules (M1)–(M6) of the box at the beginning of these exercises.

1. Each entry is r, c of the residual arc:

arc (i, j)	(1, 3)	(2, 3)	(2, 5)	(4, 2)	(4, 3)	(4, 5)	(5, 1)	(5, 3)
$\tilde{x}_{ij}$	0	0	0	0	2	4	0	3
forward (i, j) : r, c	1, 1	1, 8	2, 6	3, 4	–	–	1, 7	–
backward (j, i) : r, c	–	–	–	–	2, –4	4, –6	–	3, –3

$G(\tilde{\mathbf{x}})$ has 8 arcs.

2. Each row is a pass; next to a label that decreased, its new predecessor:

k	$d^k(1)$	$d^k(2)$	$d^k(3)$	$d^k(4)$	$d^k(5)$
0	0	0	0	0	0
1	0	0	0	-6 (5)	-3 (3)
2	0	-2 (4)	0	-9 (5)	-3
3	0	-5 (4)	0	-9	-3
4	0	-5	0	-9	-3

Pass 4 repeats the labels of pass 3: $G(\tilde{x})$ has no negative cycle, and by (M2) $\tilde{x}$ is optimal, of cost 41 (Gurobi confirms it).

3. With $\pi_i = -d^4(i)$, shifted so that $\pi_5 = 0$: $\pi = (-3, 2, -3, 6, 0)$. The reduced costs are

arc (i, j)	(1, 3)	(2, 3)	(2, 5)	(4, 2)	(4, 3)	(4, 5)	(5, 1)	(5, 3)
$\tilde{x}_{ij}$	0	0	0	0	2	4	0	3
u_{ij}	1	1	2	3	2	4	1	3
$\bar{c}_{ij}$	1	3	4	0	-5	0	4	0

- the arcs with $\bar{c}_{ij} > 0$, (1, 3), (2, 3), (2, 5), (5, 1), are empty;
- the arcs with $\bar{c}_{ij} < 0$, (4, 3), are saturated;
- no arc is strictly between its bounds.

The conditions of (M2) hold on every arc: π certifies the optimality of $\tilde{x}$.

Bellman–Ford on $G(\tilde{x})$ stops after 4 passes without negative cycles; $\pi = (-3, 2, -3, 6, 0)$ gives reduced costs that satisfy (M2), and $\tilde{x}$ is optimal with cost 41.

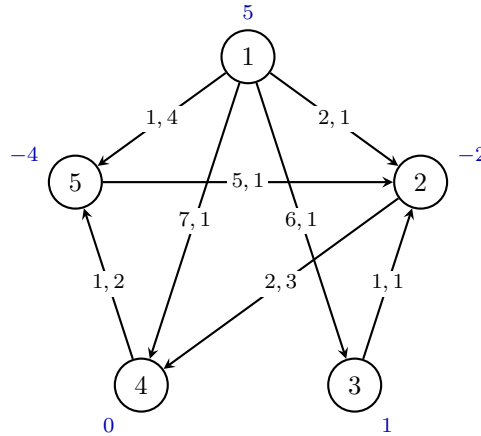
Exercise 6.3 — The cycle-canceling algorithm

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 5\}$, the 8 arcs

$$A = \{(1, 2), (1, 3), (1, 4), (1, 5), (2, 4), (3, 2), (4, 5), (5, 2)\},$$

zero lower bounds and the balances $b_1 = 5$, $b_2 = -2$, $b_3 = 1$, $b_4 = 0$, $b_5 = -4$. Costs and capacities are given in the table and, as c_{ij}, u_{ij} , in the figure; the balances are next to the nodes. The table also gives a feasible flow x^0 .

arc (i, j)	(1, 2)	(1, 3)	(1, 4)	(1, 5)	(2, 4)	(3, 2)	(4, 5)	(5, 2)
c_{ij}	2	6	7	1	2	1	1	5
u_{ij}	1	1	1	4	3	1	2	1
x_{ij}^0	1	0	1	3	1	1	2	1



Questions

1. Run the cycle-canceling algorithm from $\mathbf{x}^0$ with the rules (M3) and (M4). For every iteration give the residual network, the table of Bellman–Ford, the cycle and the new flow.
2. Give the optimal flow, its cost and the potentials that certify it.
3. How many iterations does the bound $\mathbf{c}'\mathbf{x}^0 - z(\text{MCF})$ allow?

Solution

We use the rules (M1)–(M6) of the box at the beginning of these exercises.

1. Iteration 1

The flow costs 22, and $G(\mathbf{x}^0)$ is

arc (i, j)	(1, 2)	(1, 3)	(1, 4)	(1, 5)	(2, 4)	(3, 2)	(4, 5)	(5, 2)
x_{ij}^0	1	0	1	3	1	1	2	1
forward $(i, j): r, c$	–	1, 6	–	1, 1	2, 2	–	–	–
backward $(j, i): r, c$	1, –2	–	1, –7	3, –1	1, –2	1, –1	2, –1	1, –5

Bellman–Ford gives (in bold the entries read backwards):

k	$d^k(1)$	$d^k(2)$	$d^k(3)$	$d^k(4)$	$d^k(5)$
0	0	0	0	0	0
1	–7 (4)	–2 (4)	–1 (2)	–1 (5)	–5 (2)
2	–8 (4)	–3 (4)	–3 (2)	–6 (5)	–7 (2)
3	–13 (4)	–8 (4)	–4 (2)	–8 (5)	–8 (2)
4	–15 (4)	–10 (4)	–9 (2)	–9 (5)	–13 (2)
5	–16 (4)	–11 (4)	–11 (2)	–14 (5)	–15 (2)

In pass 5 the smallest node whose label decreases is $j = 1$; the walk is 5, 4, 2, 5, 4, 1, and the cycle is $W = 5, 4, 2, 5$, with the arcs (5, 4) backward of (4, 5), (4, 2) backward of (2, 4), (2, 5) backward of (5, 2):

$$c(W) = -1 - 2 - 5 = -8, \quad r(W) = \min\{2, 1, 1\} = 1.$$

Augmenting by $\theta = 1$ gives the flow $\mathbf{x}^1$ of cost $22 + 1 \cdot (-8) = 14$.

Iteration 2

The flow costs 14, and $G(\mathbf{x}^1)$ is

arc (i, j)	(1, 2)	(1, 3)	(1, 4)	(1, 5)	(2, 4)	(3, 2)	(4, 5)	(5, 2)
x_{ij}^1	1	0	1	3	0	1	1	0
forward (i, j) : r, c	–	1, 6	–	1, 1	3, 2	–	1, 1	1, 5
backward (j, i) : r, c	1, –2	–	1, –7	3, –1	–	1, –1	1, –1	–

Bellman–Ford gives (in bold the entries read backwards):

k	$d^k(1)$	$d^k(2)$	$d^k(3)$	$d^k(4)$	$d^k(5)$
0	0	0	0	0	0
1	–7 (4)	0	–1 (2)	–1 (5)	0
2	–8 (4)	0	–1	–1	–6 (1)
3	–8	–1 (5)	–2 (1)	–7 (5)	–7 (1)
4	–14 (4)	–2 (5)	–2	–8 (5)	–7
5	–15 (4)	–2	–8 (1)	–8	–13 (1)

In pass 5 the smallest node whose label decreases is $j = 1$; the walk is 5, 4, 1, 5, 4, 1, and the cycle is $W = 5, 4, 1, 5$, with the arcs (5, 4) backward of (4, 5), (4, 1) backward of (1, 4), (1, 5) forward:

$$c(W) = -1 - 7 + 1 = -7, \quad r(W) = \min\{1, 1, 1\} = 1.$$

Augmenting by $\theta = 1$ gives the flow $\mathbf{x}^2$ of cost $14 + 1 \cdot (-7) = 7$.

Iteration 3

$G(\mathbf{x}^2)$ is

arc (i, j)	(1, 2)	(1, 3)	(1, 4)	(1, 5)	(2, 4)	(3, 2)	(4, 5)	(5, 2)
x_{ij}^2	1	0	0	4	0	1	0	0
forward (i, j) : r, c	–	1, 6	1, 7	–	3, 2	–	2, 1	1, 5
backward (j, i) : r, c	1, –2	–	–	4, –1	–	1, –1	–	–

and Bellman–Ford gives

k	$d^k(1)$	$d^k(2)$	$d^k(3)$	$d^k(4)$	$d^k(5)$
0	0	0	0	0	0
1	-2 (2)	0	-1 (2)	0	0
2	-2	0	-1	0	0

Pass 2 repeats the labels: no negative cycle, **STOP**.

2. The last flow is optimal:

arc (i, j)	(1, 2)	(1, 3)	(1, 4)	(1, 5)	(2, 4)	(3, 2)	(4, 5)	(5, 2)
$\tilde{x}_{ij}$	1	0	0	4	0	1	0	0

with cost $z(\text{MCF}) = 7$ (Gurobi confirms it). The labels of the last table give, shifted so that $\pi_5 = 0$, the potentials $\boldsymbol{\pi} = (2, 0, 1, 0, 0)$; the reduced costs of the arcs of the last residual network are all non-negative, as (M2) requires.

3. The cost goes from 22 to 7 and every iteration lowers it by at least 1: at most $22 - 7 = 15$ iterations. The algorithm needs 2, lowering the cost by 8 and 7.

Two cycles are canceled, 5, 4, 2, 5 and 5, 4, 1, 5; the cost goes from 22 to 14 and 7, the optimal value, certified by $\boldsymbol{\pi} = (2, 0, 1, 0, 0)$.

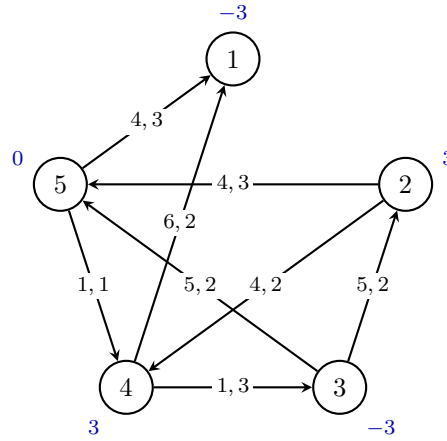
Exercise 6.4 — The successive shortest path algorithm

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 5\}$, the 8 arcs

$$A = \{(2, 4), (2, 5), (3, 2), (3, 5), (4, 1), (4, 3), (5, 1), (5, 4)\},$$

zero lower bounds and the balances $b_1 = -3, b_2 = 3, b_3 = -3, b_4 = 3, b_5 = 0$. Costs and capacities are given in the table and, as c_{ij}, u_{ij} , in the figure; the balances are next to the nodes.

arc (i, j)	(2, 4)	(2, 5)	(3, 2)	(3, 5)	(4, 1)	(4, 3)	(5, 1)	(5, 4)
c_{ij}	4	4	5	5	6	1	4	1
u_{ij}	2	3	2	2	2	3	3	1



Questions

1. Run the successive shortest path algorithm with the rules (M6). For every iteration give s , the reduced costs, the distances of Dijkstra's algorithm, t , the path P , δ and the new potentials.
2. Give the optimal flow and its cost, and check its optimality with the final potentials.
3. Compare the number of iterations with the bound $D = \sum_{i:b_i>0} b_i$.

Solution

We use the rules (M1)–(M6) of the box at the beginning of these exercises.

1. We start from $\mathbf{x} = \mathbf{0}$ and $\boldsymbol{\pi} = \mathbf{0}$.

Iteration 1

$s = 2$, with $\boldsymbol{\pi} = (0, 0, 0, 0, 0)$ and $\mathbf{e} = (-3, 3, -3, 3, 0)$. The reduced costs of the arcs of A (forward arc: $\bar{c}_{ij}$ if $x_{ij} < u_{ij}$; backward arc: $-\bar{c}_{ij}$ if $x_{ij} > 0$) are

arc (i, j)	(2, 4)	(2, 5)	(3, 2)	(3, 5)	(4, 1)	(4, 3)	(5, 1)	(5, 4)
x_{ij}	0	0	0	0	0	0	0	0
$\bar{c}_{ij}$	4	4	5	5	6	1	4	1

Dijkstra from s makes the nodes permanent in the order 2,4,5,3,1, with distances $d = (8, 0, 5, 4, 4)$. The deficit nodes are 1, 3: $t = 3$, with the path $P = 2, 4, 3$ of reduced cost 5 and cost $c(P) = 5$. Then $\delta = \min\{3, 3, 2, 3\} = 2$, the new potentials are $\boldsymbol{\pi} = (-5, 0, -5, -4, -4)$, and the cost becomes 10.

Iteration 2

$s = 2$, with $\boldsymbol{\pi} = (-5, 0, -5, -4, -4)$ and $\mathbf{e} = (-3, 1, -1, 3, 0)$. The reduced costs of the arcs of A (forward arc: $\bar{c}_{ij}$ if $x_{ij} < u_{ij}$; backward arc: $-\bar{c}_{ij}$ if $x_{ij} > 0$) are

arc (i, j)	(2, 4)	(2, 5)	(3, 2)	(3, 5)	(4, 1)	(4, 3)	(5, 1)	(5, 4)
x_{ij}	2	0	0	0	0	2	0	0
$\bar{c}_{ij}$	0	0	10	6	5	0	3	1

Dijkstra from s makes the nodes permanent in the order 2,5,4,3,1, with distances $d = (3, 0, 1, 1, 0)$. The deficit nodes are 1, 3: $t = 3$, with the path $P = 2, 5, 4, 3$ of reduced cost 1 and cost $c(P) = 6$. Then $\delta = \min\{1, 1, 3, 1, 1\} = 1$, the new potentials are $\pi = (-6, 0, -6, -5, -4)$, and the cost becomes 16.

Iteration 3

$s = 4$, with $\pi = (-6, 0, -6, -5, -4)$ and $e = (-3, 0, 0, 3, 0)$. The reduced costs of the arcs of A (forward arc: $\bar{c}_{ij}$ if $x_{ij} < u_{ij}$; backward arc: $-\bar{c}_{ij}$ if $x_{ij} > 0$) are

arc (i, j)	(2, 4)	(2, 5)	(3, 2)	(3, 5)	(4, 1)	(4, 3)	(5, 1)	(5, 4)
x_{ij}	2	1	0	0	0	3	0	1
$\bar{c}_{ij}$	-1	0	11	7	5	0	2	0

Dijkstra from s makes the nodes permanent in the order 4,5,2,1, with distances $d = (2, 0, +\infty, 0, 0)$. The deficit node is 1: $t = 1$, with the path $P = 4, 5, 1$ of reduced cost 2 and cost $c(P) = 3$. Then $\delta = \min\{3, 3, 1, 3\} = 1$, the new potentials are $\pi = (-8, 0, -8, -5, -4)$, and the cost becomes 19.

Iteration 4

$s = 4$, with $\pi = (-8, 0, -8, -5, -4)$ and $e = (-2, 0, 0, 2, 0)$. The reduced costs of the arcs of A (forward arc: $\bar{c}_{ij}$ if $x_{ij} < u_{ij}$; backward arc: $-\bar{c}_{ij}$ if $x_{ij} > 0$) are

arc (i, j)	(2, 4)	(2, 5)	(3, 2)	(3, 5)	(4, 1)	(4, 3)	(5, 1)	(5, 4)
x_{ij}	2	1	0	0	0	3	1	0
$\bar{c}_{ij}$	-1	0	13	9	3	-2	0	0

Dijkstra from s makes the nodes permanent in the order 4,2,5,1, with distances $d = (1, 1, +\infty, 0, 1)$. The deficit node is 1: $t = 1$, with the path $P = 4, 2, 5, 1$ of reduced cost 1 and cost $c(P) = 4$. Then $\delta = \min\{2, 2, 2, 2, 2\} = 2$, the new potentials are $\pi = (-9, -1, -9, -5, -5)$, and the cost becomes 27.

Now $e = \mathbf{0}$: STOP.

2. The final flow is

arc (i, j)	(2, 4)	(2, 5)	(3, 2)	(3, 5)	(4, 1)	(4, 3)	(5, 1)	(5, 4)
$\tilde{x}_{ij}$	0	3	0	0	0	3	3	0

with cost 27 (Gurobi confirms it). Shifted so that $\pi_5 = 0$, the final potentials are $\pi = (-4, 4, -4, 0, 0)$, with reduced costs

arc (i, j)	(2, 4)	(2, 5)	(3, 2)	(3, 5)	(4, 1)	(4, 3)	(5, 1)	(5, 4)
$\bar{c}_{ij}$	0	0	13	9	2	-3	0	1

Every arc with $\bar{c}_{ij} > 0$ is empty and every arc with $\bar{c}_{ij} < 0$ is saturated: (M2) holds.

- The total supply is $D = 6$ and the algorithm needs 4 iterations, since the amounts δ are 2, 1, 1, 2, which add up to 6.

The successive shortest path algorithm sends the 6 units along 4 paths, of costs 5, 6, 3, 4; the optimal cost is 27.

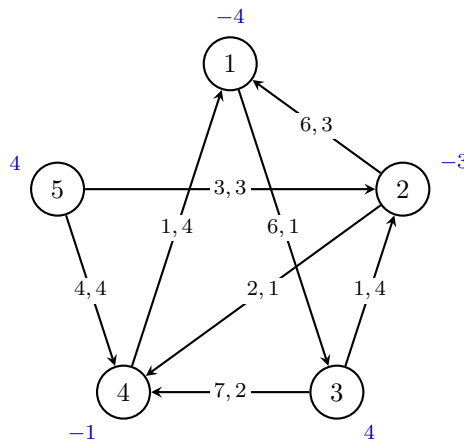
Exercise 6.5 — A first feasible flow and the decomposition

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 5\}$, the 8 arcs

$$A = \{(1, 3), (2, 1), (2, 4), (3, 2), (3, 4), (4, 1), (5, 2), (5, 4)\},$$

zero lower bounds and the balances $b_1 = -4$, $b_2 = -3$, $b_3 = 4$, $b_4 = -1$, $b_5 = 4$. Costs and capacities are given in the table and, as c_{ij}, u_{ij} , in the figure; the balances are next to the nodes. The table also gives the optimal flow $\tilde{x}$, computed with an LP solver.

arc (i, j)	(1, 3)	(2, 1)	(2, 4)	(3, 2)	(3, 4)	(4, 1)	(5, 2)	(5, 4)
c_{ij}	6	6	2	1	7	1	3	4
u_{ij}	1	3	1	4	2	4	3	4
$\tilde{x}_{ij}$	0	0	1	4	0	4	0	4



Questions

- Build the auxiliary network of (M5) and compute a maximum flow with the Edmonds–Karp algorithm. Does the network have a feasible flow? Give the flow x^0 on the arcs of A and its cost.

2. Write the circulation $\mathbf{y}$ on $G(\mathbf{x}^0)$ such that $\tilde{\mathbf{x}}$ is $\mathbf{x}^0$ augmented by $\mathbf{y}$, and decompose it into cycles with the rule of (M5).
3. Check that $\sum_h \theta_h c(W_h) = \mathbf{c}'\tilde{\mathbf{x}} - \mathbf{c}'\mathbf{x}^0$. Why does a decomposition of $\tilde{\mathbf{x}} - \mathbf{x}$ contain a negative cycle whenever a feasible flow $\mathbf{x}$ is not optimal?

Solution

We use the rules (M1)–(M6) of the box at the beginning of these exercises.

1. The new arcs are $(s, 3)$ with capacity 4, $(s, 5)$ with capacity 4, $(1, t)$ with capacity 4, $(2, t)$ with capacity 3, $(4, t)$ with capacity 1, and $D = 8$. The augmenting paths of the Edmonds–Karp algorithm are

path	nodes	units
1	$s, 3, 2, t$	3
2	$s, 3, 4, t$	1
3	$s, 5, 2, 1, t$	3
4	$s, 5, 4, 1, t$	1

The value is $8 = D$: the network has a feasible flow, the restriction to A

arc (i, j)	$(1, 3)$	$(2, 1)$	$(2, 4)$	$(3, 2)$	$(3, 4)$	$(4, 1)$	$(5, 2)$	$(5, 4)$
x_{ij}^0	0	3	0	3	1	1	3	1

with cost 42.

2. By the lemma on the difference of two feasible flows, $\mathbf{y}$ is positive on the forward arc (i, j) with $y_{ij} = \tilde{x}_{ij} - x_{ij}^0$ when $\tilde{x}_{ij} > x_{ij}^0$, and on the backward arc (j, i) with $y_{ji} = x_{ij}^0 - \tilde{x}_{ij}$ when $\tilde{x}_{ij} < x_{ij}^0$:

$$y_{12} = 3, \quad y_{24} = 1, \quad y_{25} = 3, \quad y_{32} = 1, \quad y_{41} = 3, \quad y_{43} = 1, \quad y_{54} = 3.$$

Starting from the first arc with $y > 0$ and leaving every node along its arc with $y > 0$ of smallest head:

h	cycle W_h	θ_h	$c(W_h)$
1	$1, 2, 4, 1$	1	$-6 + 2 + 1 = -3$
2	$1, 2, 5, 4, 1$	2	$-6 - 3 + 4 + 1 = -4$
3	$2, 5, 4, 3, 2$	1	$-3 + 4 - 7 + 1 = -5$

3. Indeed

$$\sum_h \theta_h c(W_h) = 1 \cdot (-3) + 2 \cdot (-4) + 1 \cdot (-5) = -16 = 26 - 42.$$

If $\mathbf{x}$ is not optimal, $\mathbf{c}'\tilde{\mathbf{x}} - \mathbf{c}'\mathbf{x} < 0$: a sum of terms $\theta_h c(W_h)$ with $\theta_h > 0$ is negative only if some $c(W_h)$ is negative. This is the argument behind the negative cycle optimality conditions.

The maximum flow has value 8 and gives a feasible flow of cost 42; the difference with the optimal flow, of cost 26, splits into 3 cycles of $G(\mathbf{x}^0)$ whose costs add up to -16 .

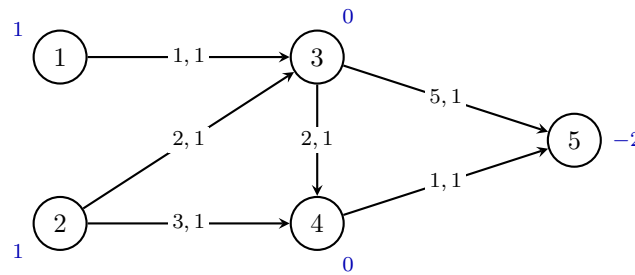
Exercise 6.6 — A counterexample and a proof

We consider the network $G = (N, A)$ with $N = \{1, 2, \dots, 5\}$, the 6 arcs

$$A = \{(1, 3), (2, 3), (2, 4), (3, 4), (3, 5), (4, 5)\},$$

zero lower bounds and the balances $b_1 = 1, b_2 = 1, b_3 = 0, b_4 = 0, b_5 = -2$. Costs and capacities are given in the table and, as c_{ij}, u_{ij} , in the figure; the balances are next to the nodes.

arc (i, j)	$(1, 3)$	$(2, 3)$	$(2, 4)$	$(3, 4)$	$(3, 5)$	$(4, 5)$
c_{ij}	1	2	3	2	5	1
u_{ij}	1	1	1	1	1	1



In the successive shortest path algorithm, a student runs Dijkstra's algorithm with the costs c_{ij}^x of the residual network as lengths, instead of the reduced costs.

Questions

1. Run the first iteration of the successive shortest path algorithm with the rules (M6).
2. In the second iteration, run Dijkstra's algorithm from s with the lengths c_{ij}^x , as the student does. Compare its labels with the true minimum costs of the paths from s , and give the flow that the student obtains and its cost.
3. Complete the second iteration with the reduced costs, and give the optimal flow.
4. Prove: if $\tilde{\mathbf{x}}$ is an optimal flow and every cycle of $G(\tilde{\mathbf{x}})$ has positive cost, then $\tilde{\mathbf{x}}$ is the only optimal flow.

Solution

We use the rules (M1)–(M6) of the box at the beginning of these exercises.

1. We start from $\mathbf{x} = \mathbf{0}$ and $\boldsymbol{\pi} = \mathbf{0}$.

Iteration 1

$s = 1$, with $\pi = (0, 0, 0, 0, 0)$ and $e = (1, 1, 0, 0, -2)$. The reduced costs of the arcs of A (forward arc: $\bar{c}_{ij}$ if $x_{ij} < u_{ij}$; backward arc: $-\bar{c}_{ij}$ if $x_{ij} > 0$) are

arc (i, j)	(1, 3)	(2, 3)	(2, 4)	(3, 4)	(3, 5)	(4, 5)
x_{ij}	0	0	0	0	0	0
$\bar{c}_{ij}$	1	2	3	2	5	1

Dijkstra from s makes the nodes permanent in the order 1,3,4,5, with distances $d = (0, +\infty, 1, 3, 4)$. The deficit node is 5: $t = 5$, with the path $P = 1, 3, 4, 5$ of reduced cost 4 and cost $c(P) = 4$. Then $\delta = \min\{1, 2, 1, 1, 1\} = 1$, the new potentials are $\pi = (0, -4, -1, -3, -4)$, and the cost becomes 4.

Node 1 has sent its unit; now $e_2 > 0$, so the second iteration starts from $s = 2$.

2. After the first iteration the residual network is

arc (i, j)	(1, 3)	(2, 3)	(2, 4)	(3, 4)	(3, 5)	(4, 5)
x_{ij}^1	1	0	0	1	0	1
forward $(i, j): r, c$	–	1, 2	1, 3	–	1, 5	–
backward $(j, i): r, c$	1, –1	–	–	1, –2	–	1, –1

and its arcs (3, 1), (4, 3), (5, 4) have negative cost. Dijkstra with these lengths makes the nodes permanent in the order 2, 3, 1, 4, 5 and gives the labels (1, 0, 2, 3, 7). The true minimum costs from 2, computed with Bellman–Ford ($G(\mathbf{x}^1)$ has no negative cycle), are (0, 0, 1, 3, 6).

- Node 3 becomes permanent with label 2; later node 4, with label 3, offers $3 - 2 = 1$ along the negative arc (4, 3), but node 3 is already permanent and is not updated.
- Hence the label of $t = 5$ is 7, through the path 2, 3, 5, instead of 6, the cost of 2, 4, 3, 5.

Augmenting along 2, 3, 5 by $\delta = 1$ the student obtains the feasible flow

arc (i, j)	(1, 3)	(2, 3)	(2, 4)	(3, 4)	(3, 5)	(4, 5)
x_{ij}	1	1	0	1	1	1

of cost 11, which is not optimal.

3. With the reduced costs all the lengths are non-negative:

Iteration 2

$s = 2$, with $\pi = (0, -4, -1, -3, -4)$ and $e = (0, 1, 0, 0, -1)$. The reduced costs of the arcs of A (forward arc: $\bar{c}_{ij}$ if $x_{ij} < u_{ij}$; backward arc: $-\bar{c}_{ij}$ if $x_{ij} > 0$) are

arc (i, j)	(1, 3)	(2, 3)	(2, 4)	(3, 4)	(3, 5)	(4, 5)
x_{ij}	1	0	0	1	0	1
$\bar{c}_{ij}$	0	5	4	0	2	0

Dijkstra from s makes the nodes permanent in the order 2,4,3,1,5, with distances $d = (4, 0, 4, 4, 6)$. The deficit node is 5: $t = 5$, with the path $P = 2, 4, 3, 5$ of reduced cost 6 and cost $c(P) = 6$. Then $\delta = \min\{1, 1, 1, 1, 1\} = 1$, the new potentials are $\pi = (-4, -4, -5, -7, -10)$, and the cost becomes 10.

Now $e = \mathbf{0}$, and the optimal flow is

arc (i, j)	(1, 3)	(2, 3)	(2, 4)	(3, 4)	(3, 5)	(4, 5)
$\tilde{x}_{ij}$	1	0	1	0	1	1

of cost 10 (Gurobi confirms it). The path of the second iteration costs $c(P) = 6$, the true minimum cost.

- Let $\mathbf{x}'$ be an optimal flow. By the lemma on the difference of two feasible flows, $\mathbf{c}'\mathbf{x}' - \mathbf{c}'\tilde{\mathbf{x}} = \sum_{h=1}^k \theta_h c(W_h)$ for a decomposition of the circulation $\mathbf{y}$ into cycles W_h of $G(\tilde{\mathbf{x}})$ with $\theta_h > 0$. The left-hand side is 0, since both flows are optimal, and every term on the right is positive by assumption. Hence the sum has no terms: $\mathbf{y} = \mathbf{0}$ and $\mathbf{x}' = \tilde{\mathbf{x}}$.

With the costs c_{ij}^x , Dijkstra's algorithm gives node 5 the label 7 instead of 6, and the student ends with cost 11; the reduced costs make the lengths non-negative, and the algorithm ends with the optimal cost 10.

Transportation and assignment

Rules used in these exercises

The definitions and the results of the chapter «[Transportation and assignment](#)». The origins are $1, 2, \dots, m$ with supplies a_i , the destinations $m+1, m+2, \dots, m+n$ with demands d_j , and the cells (i, j) are taken in row-column order.

- (F1) **Model.** $\min \sum_i \sum_j c_{ij} x_{ij}$ with row sums a_i , column sums d_j and $\mathbf{x} \geq \mathbf{0}$; feasible if and only if $\sum_i a_i = \sum_j d_j$. An excess of supply goes to a dummy destination with costs 0.
- (F2) **Bases.** A basis is a set of $m+n-1$ cells forming a spanning tree of the bipartite network (no closed path alternating rows and columns). Its basic solution is found leaf by leaf; it is degenerate if a basic cell has value 0.
- (F3) **Initial solutions.** North-west corner: the active cell with the smallest row and column. Minimum cost: the cheapest active cell, the smallest i and then j on ties. The amount is $\min\{\bar{a}_i, \bar{d}_j\}$; if $\bar{a}_i = 0$ and another row is active, the row is crossed out, otherwise the column.
- (F4) **Potential method.** $\pi_i - \pi_j = c_{ij}$ on the basic cells, $\pi_{m+n} = 0$; $\bar{c}_{ij} = c_{ij} - \pi_i + \pi_j$. Optimal if $\bar{c} \geq 0$, unique if $\bar{c}_{ij} > 0$ outside the basis. Otherwise the most negative $\bar{c}_e$ enters (ties: smallest i , then j); the cycle starts at e along its row with signs $+, -, +, \dots$; θ is the minimum on the cells $-$, and the first of them with $\bar{x} = \theta$ in row-column order leaves; the cost changes by $\theta \bar{c}_e$.
- (F5) **Hungarian algorithm.** Subtract the row minima (π_i) and then the column minima ($-\pi_j$). Find a maximum set of independent zeros (row by row, alternating paths, columns in increasing order); if it has n zeros, stop. Otherwise the lines are the rows not reached and the columns reached by alternating paths from the unmatched rows; δ is the minimum uncovered entry, subtracted from the uncovered entries and added to those covered twice ($\pi_i \pm \delta$ on uncovered rows, $\pi_j \pm \delta$ on covered columns). The dual value $\sum_i \pi_i - \sum_j \pi_j$ grows by $\delta(n-k)$ with k lines.

Exercise 7.1 — Initial basic feasible solutions

Three plants 1, 2, 3 supply $a_1 = 30$, $a_2 = 40$, $a_3 = 20$ units of a product, and three customers 4, 5, 6 demand $d_4 = 15$, $d_5 = 25$, $d_6 = 40$ units. Every demand must be met, and every plant ships at most its supply. The unit shipping costs are:

c_{ij}	4	5	6	a_i
1	7	4	7	30
2	1	6	8	40
3	9	9	6	20
d_j	15	25	40	

Questions

1. Is the instance balanced? Add a dummy destination as in (F1) and write the transportation tableau.
2. Build an initial basic feasible solution with the north-west corner rule: list the steps, write the solution in the tableau and compute its cost.
3. Build an initial basic feasible solution with the minimum cost rule, and compare the two costs.
4. Check that both solutions have $m + n - 1$ basic cells forming a spanning tree. How much supply does each plant leave unused in the two solutions?

Solution

We use the rules (F1)–(F5) of the box at the beginning of these exercises.

1. The total supply is $30 + 40 + 20 = 90$ and the total demand is $15 + 25 + 40 = 80$: the instance is not balanced, with an excess of 10 units. We add the dummy destination 7 with demand $d_7 = 10$ and costs $c_{17} = c_{27} = c_{37} = 0$:

c_{ij}	4	5	6	7	a_i
1	7	4	7	0	30
2	1	6	8	0	40
3	9	9	6	0	20
d_j	15	25	40	10	90

2. The steps of the north-west corner rule are

step	cell	amount	crossed out
1	(1, 4)	15	column 4
2	(1, 5)	15	row 1
3	(2, 5)	10	column 5
4	(2, 6)	30	row 2
5	(3, 6)	10	column 6
6	(3, 7)	10	row 3, column 7

and the solution is

$\bar{x}_{ij}$	4	5	6	7	a_i
1	15	15			30
2		10	30		40
3			10	10	20
d_j	15	25	40	10	

Its cost is $7 \cdot 15 + 4 \cdot 15 + 6 \cdot 10 + 8 \cdot 30 + 6 \cdot 10 + 0 \cdot 10 = 525$.

3. The steps of the minimum cost rule are

step	cell	amount	crossed out
1	(1, 7)	10	column 7
2	(2, 4)	15	column 4
3	(1, 5)	20	row 1
4	(2, 5)	5	column 5
5	(3, 6)	20	row 3
6	(2, 6)	20	row 2, column 6

At step 1 the three dummy cells cost 0, and the rule takes (1, 7), the smallest row. The solution is

$\bar{x}_{ij}$	4	5	6	7	a_i
1		20		10	30
2	15	5	20		40
3			20		20
d_j	15	25	40	10	

Its cost is $4 \cdot 20 + 0 \cdot 10 + 1 \cdot 15 + 6 \cdot 5 + 8 \cdot 20 + 6 \cdot 20 = 405$, lower than the 525 of the north-west corner rule.

4. Each rule makes $m + n - 1 = 3 + 4 - 1 = 6$ steps, one cell per step; the cells form a spanning tree because the crossed line of each step contains no later cell. The flow on the dummy column is the unused supply: with the north-west corner rule plant 1 keeps 0, plant 2 keeps 0 and plant 3 keeps 10; with the minimum cost rule plant 1 keeps 10, plant 2 keeps 0 and plant 3 keeps 0.

With the dummy destination 7 ($d_7 = 10$, zero costs) the north-west corner rule costs 525 and the minimum cost rule 405.

Exercise 7.2 — The potential method

We consider the transportation instance with origins 1, 2, 3, destinations 4, 5, 6, 7, and the costs, supplies and demands of the tableau:

c_{ij}	4	5	6	7	a_i
1	4	6	8	8	30
2	2	2	7	7	15
3	8	7	3	2	25
d_j	15	5	45	5	70

Questions

1. Build the initial basic feasible solution with the north-west corner rule and compute its cost.
2. Compute the potentials of its basis ($\pi_7 = 0$) and the reduced costs. Is the solution optimal?
3. Run the potential method until the optimal solution, with the rules of (F4).
4. Write the optimal potentials and check that the value of the dual model equals the optimal cost. Is the optimal solution unique?

Solution

We use the rules (F1)–(F5) of the box at the beginning of these exercises.

1. The north-west corner rule gives

step	cell	amount	crossed out
1	(1, 4)	15	column 4
2	(1, 5)	5	column 5
3	(1, 6)	10	row 1
4	(2, 6)	15	row 2
5	(3, 6)	20	column 6
6	(3, 7)	5	row 3, column 7

$\bar{x}_{ij}$	4	5	6	7	a_i
1	15	5	10		30
2			15		15
3			20	5	25
d_j	15	5	45	5	

with cost $4 \cdot 15 + 6 \cdot 5 + 8 \cdot 10 + 7 \cdot 15 + 3 \cdot 20 + 2 \cdot 5 = 345$.

- From $\pi_7 = 0$ along the basic cells: $\pi_1 = 7$, $\pi_2 = 6$, $\pi_3 = 2$, $\pi_4 = 3$, $\pi_5 = 1$, $\pi_6 = -1$, $\pi_7 = 0$. The reduced costs of the non-basic cells are in the first tableau of the next answer; $\bar{c}_{25} = -3$ and $\bar{c}_{24} = -1$ are negative, so the solution is not optimal.
- The tableaux contain the amounts of the basic cells (blue, bold) and the reduced costs of the other cells; the potentials are on the margins.

Iteration 1

Potentials: $\pi_1 = 7$, $\pi_2 = 6$, $\pi_3 = 2$, $\pi_4 = 3$, $\pi_5 = 1$, $\pi_6 = -1$, $\pi_7 = 0$.

	4	5	6	7	π_i
1	15	5 $\ominus$	10 $\oplus$	1	7
2	-1	-3 $\oplus$	15 $\ominus$	1	6
3	9	6	20	5	2
π_j	3	1	-1	0	

- The negative reduced costs are $\bar{c}_{25} = -3$, $\bar{c}_{24} = -1$; the cell (2, 5) enters the basis.
- The cycle is $(2, 5)^+$, $(2, 6)^-$, $(1, 6)^+$, $(1, 5)^-$.
- The cells with sign $-$ have $\bar{x}_{26} = 15$ and $\bar{x}_{15} = 5$: $\theta = 5$, and the cell (1, 5) leaves the basis.
- The cost goes from 345 to $345 + 5 \cdot (-3) = 330$.

Iteration 2

Potentials: $\pi_1 = 7, \pi_2 = 6, \pi_3 = 2, \pi_4 = 3, \pi_5 = 4, \pi_6 = -1, \pi_7 = 0$.

	4	5	6	7	π_i
1	15 $\ominus$	3	15 $\oplus$	1	7
2	-1 $\oplus$	5	10 $\ominus$	1	6
3	9	9	20	5	2
π_j	3	4	-1	0	

- The only negative reduced cost is $\bar{c}_{24} = -1$: the cell $(2,4)$ enters the basis.
- The cycle is $(2,4)^+, (2,6)^-, (1,6)^+, (1,4)^-$.
- The cells with sign $-$ have $\bar{x}_{26} = 10$ and $\bar{x}_{14} = 15$: $\theta = 10$, and the cell $(2,6)$ leaves the basis.
- The cost goes from 330 to $330 + 10 \cdot (-1) = 320$.

Iteration 3

Potentials: $\pi_1 = 7, \pi_2 = 5, \pi_3 = 2, \pi_4 = 3, \pi_5 = 3, \pi_6 = -1, \pi_7 = 0$.

	4	5	6	7	π_i
1	5	2	25	1	7
2	10	5	1	2	5
3	9	8	20	5	2
π_j	3	3	-1	0	

- All the reduced costs are non-negative: the basic feasible solution is optimal, with cost $z = 320$.
4. The optimal potentials are $\pi_1 = 7, \pi_2 = 5, \pi_3 = 2, \pi_4 = 3, \pi_5 = 3, \pi_6 = -1, \pi_7 = 0$, and the dual value is

$$\sum_i a_i \pi_i - \sum_j d_j \pi_j = 335 - 15 = 320,$$

equal to the optimal cost. Every non-basic cell has $\bar{c}_{ij} > 0$, so the optimal solution is unique.

From the north-west corner (cost 345) the potential method makes 2 pivots and reaches the unique optimum $\tilde{x}_{14} = 5, \tilde{x}_{16} = 25, \tilde{x}_{24} = 10, \tilde{x}_{25} = 5, \tilde{x}_{36} = 20, \tilde{x}_{37} = 5$, with cost 320.

Exercise 7.3 — Degeneracy

We consider the transportation instance with origins 1, 2, 3 and destinations 4, 5, 6:

c_{ij}	4	5	6	a_i
1	7	3	7	20
2	5	3	1	10
3	6	7	8	25
d_j	10	10	35	55

Questions

1. Build the north-west corner solution. At which step are a row and a column exhausted at the same time? Which basic cell has value 0, and why must it stay in the basis?
2. Run the potential method from this solution, with the rules of (F4).
3. Which pivot is degenerate, and what changes in it? Give the optimal solution and its cost.

Solution

We use the rules (F1)–(F5) of the box at the beginning of these exercises.

1. The north-west corner rule gives

step	cell	amount	crossed out
1	(1, 4)	10	column 4
2	(1, 5)	10	row 1
3	(2, 5)	0	column 5
4	(2, 6)	10	row 2
5	(3, 6)	25	row 3, column 6

At step 2 the cell (1, 5) exhausts both row 1 and column 5. Only the row is crossed out, and the next cell (2, 5) of column 5 gets the amount 0:

$\bar{x}_{ij}$	4	5	6	a_i
1	10	10		20
2		0	10	10
3			25	25
d_j	10	10	35	

The solution, of cost 310, is degenerate. The basis needs $m + n - 1 = 5$ cells: without (2, 5) the other cells do not span the 6 nodes, and the potentials cannot be computed.

2. Iteration 1

Potentials: $\pi_1 = 1, \pi_2 = 1, \pi_3 = 8, \pi_4 = -6, \pi_5 = -2, \pi_6 = 0$.

	4	5	6	π_i
1	$10 \ominus$	$10 \oplus$	6	1
2	-2	$0 \ominus$	$10 \oplus$	1
3	$-8 \oplus$	-3	$25 \ominus$	8
π_j	-6	-2	0	

- The negative reduced costs are $\bar{c}_{34} = -8, \bar{c}_{35} = -3, \bar{c}_{24} = -2$; the cell (3,4) enters the basis.
- The cycle is $(3,4)^+, (3,6)^-, (2,6)^+, (2,5)^-, (1,5)^+, (1,4)^-$.
- The cells with sign $-$ have $\bar{x}_{36} = 25, \bar{x}_{25} = 0$ and $\bar{x}_{14} = 10$: $\theta = 0$, and the cell (2,5) leaves the basis.
- The pivot is degenerate: the solution and its cost $z = 310$ do not change, only the basis does.

Iteration 2

Potentials: $\pi_1 = 9, \pi_2 = 1, \pi_3 = 8, \pi_4 = 2, \pi_5 = 6, \pi_6 = 0$.

	4	5	6	π_i
1	$10 \ominus$	10	$-2 \oplus$	9
2	6	8	10	1
3	$0 \oplus$	5	$25 \ominus$	8
π_j	2	6	0	

- The only negative reduced cost is $\bar{c}_{16} = -2$: the cell (1,6) enters the basis.
- The cycle is $(1,6)^+, (1,4)^-, (3,4)^+, (3,6)^-$.
- The cells with sign $-$ have $\bar{x}_{14} = 10$ and $\bar{x}_{36} = 25$: $\theta = 10$, and the cell (1,4) leaves the basis.
- The cost goes from 310 to $310 + 10 \cdot (-2) = 290$.

Iteration 3

Potentials: $\pi_1 = 7, \pi_2 = 1, \pi_3 = 8, \pi_4 = 2, \pi_5 = 4, \pi_6 = 0$.

	4	5	6	π_i
1	2	10	10	7
2	6	6	10	1
3	10	3	15	8
π_j	2	4	0	

- All the reduced costs are non-negative: the basic feasible solution is optimal, with cost $z = 290$.
3. The first pivot has $\theta = 0$: the cell (3,4) enters with value 0 and (2,5) leaves; the solution and its cost do not change, but the new potentials show a negative reduced cost. The optimal solution is $\tilde{x}_{15} = 10$, $\tilde{x}_{16} = 10$, $\tilde{x}_{26} = 10$, $\tilde{x}_{34} = 10$, $\tilde{x}_{36} = 15$, with cost 290, and it is unique, since every non-basic cell has $\bar{c}_{ij} > 0$.

The north-west corner solution is degenerate (the basic cell (2,5) has value 0); the first pivot has $\theta = 0$, the second pivot reaches the optimum, of cost 290.

Exercise 7.4 — The Hungarian algorithm

Five workers 1, 2, 3, 4, 5 must be assigned to five jobs 6, 7, 8, 9, 10, one job each, at minimum total cost. The costs are:

c_{ij}	6	7	8	9	10
1	6	10	10	5	3
2	11	12	9	7	7
3	8	1	12	6	11
4	12	7	10	8	2
5	3	4	11	6	1

Questions

1. Reduce the rows and then the columns of the matrix. Which potentials do the reductions give, and which lower bound on the optimal cost?
2. Run the Hungarian algorithm: at each iteration give a maximum set of independent zeros, the lines of (F5) and the adjustment δ .
3. Give the optimal assignment and its cost, and check that the final potentials are an optimal dual solution.

Solution

We use the rules (F1)–(F5) of the box at the beginning of these exercises.

1. The row minima are 3, 7, 1, 2, 1, so $\pi_1 = 3$, $\pi_2 = 7$, $\pi_3 = 1$, $\pi_4 = 2$, $\pi_5 = 1$:

$c_{ij} - \pi_i$	6	7	8	9	10
1	3	7	7	2	0
2	4	5	2	0	0
3	7	0	11	5	10
4	10	5	8	6	0
5	2	3	10	5	0

The column minima are then 2, 0, 2, 0, 0, so $\pi_6 = -2$ and $\pi_8 = -2$, and the other π_j are 0. The dual value $\sum_i \pi_i - \sum_j \pi_j = 18$ is a lower bound on the cost of every assignment.

2. In each matrix the independent zeros are boxed, the lines blue and the cell of δ red; the potentials are on the margins.

Iteration 1

$\bar{c}_{ij}$	6	7	8	9	10	π_i
1	1	7	5	2	0	3
2	2	5	0	0	0	7
3	5	0	9	5	10	1
4	8	5	6	6	0	2
5	0	3	8	5	0	1
π_j	-2	0	-2	0	0	

- A maximum set of independent zeros (boxed) has 4 elements: (1, 10), (2, 8), (3, 7), (5, 6).
- The $4 < 5$ lines (blue) are the rows 2, 3 and 5 and the column 10.
- The minimum uncovered entry is $\delta = 1$, in the cell (1, 6) (red). We subtract it from the uncovered entries and add it to the entries covered twice: π_i grows by 1 for the uncovered rows 1 and 4 and π_j grows by 1 for the covered column 10. The dual value grows by $1 \cdot (5 - 4) = 1$.

Iteration 2

$\bar{c}_{ij}$	6	7	8	9	10	π_i
1	0	6	4	1	0	4
2	2	5	0	0	1	7
3	5	0	9	5	11	1
4	7	4	5	5	0	3
5	0	3	8	5	1	1
π_j	-2	0	-2	0	1	

- A maximum set of independent zeros (boxed) has 4 elements: (1, 6), (2, 8), (3, 7), (4, 10).
- The $4 < 5$ lines (blue) are the rows 2 and 3 and the columns 6 and 10.
- The minimum uncovered entry is $\delta = 1$, in the cell (1, 9) (red). We subtract it from the uncovered entries and add it to the entries covered twice: π_i grows by 1 for the uncovered rows 1, 4 and 5 and π_j grows by 1 for the covered columns 6 and 10. The dual value grows by $1 \cdot (5 - 4) = 1$.

Iteration 3

$\bar{c}_{ij}$	6	7	8	9	10	π_i
1	0	5	3	0	0	5
2	3	5	0	0	2	7
3	6	0	9	5	12	1
4	7	3	4	4	0	4
5	0	2	7	4	1	2
π_j	-1	0	-2	0	2	

- A maximum set of independent zeros (boxed) has 5 elements: (1, 9), (2, 8), (3, 7), (4, 10), (5, 6).
 - They form an assignment of zero reduced cost: the algorithm stops, and the assignment is optimal.
3. The optimal assignment is $1 \rightarrow 9, 2 \rightarrow 8, 3 \rightarrow 7, 4 \rightarrow 10, 5 \rightarrow 6$, with cost $5 + 9 + 1 + 2 + 3 = 20$. The final potentials give $\sum_i \pi_i - \sum_j \pi_j = 19 - (-1) = 20$, equal to the cost; all the reduced costs are non-negative, so the potentials are dual feasible and both solutions are optimal. The optimum is unique, as the enumeration of the 120 assignments confirms.

The reductions give the bound 18; two adjustments, with $\delta = 1$ and $\delta = 1$, lead to the optimal assignment $1 \rightarrow 9, 2 \rightarrow 8, 3 \rightarrow 7, 4 \rightarrow 10, 5 \rightarrow 6$, of cost 20.

Exercise 7.5 — Bases, cycles and the minimum cost rule

We consider the transportation instance with origins 1, 2, 3 and destinations 4, 5, 6:

c_{ij}	4	5	6	a_i
1	2	4	1	30
2	4	8	5	20
3	9	1	1	15
d_j	35	10	20	65

and the three sets of cells

$$T_1 = \{(1, 4), (2, 4), (2, 6), (3, 5), (3, 6)\}, \quad T_2 = \{(1, 6), (2, 4), (2, 6), (3, 5), (3, 6)\},$$

$$T_3 = \{(2, 4), (2, 5), (2, 6), (3, 5), (3, 6)\}.$$

Questions

1. Which of T_1 , T_2 , T_3 are bases? For each basis compute its basic solution and say whether it is feasible.
2. Prove that adding a constant k to all the costs of row i adds $k a_i$ to the cost of every feasible solution, so that the optimal solutions do not change. What happens adding k to a column j ?
3. Build the minimum cost solution and decide with the potentials whether it is optimal. Is the minimum cost rule always optimal?

Solution

We use the rules (F1)–(F5) of the box at the beginning of these exercises.

1. Each set has $m + n - 1 = 5$ cells, so it is a basis exactly when it contains no cycle (F2).
 - T_1 has no cycle: it is a spanning tree. Leaf by leaf, leaf 1 gives $x_{14} = 30$, leaf 4 gives $x_{24} = 5$, leaf 2 gives $x_{26} = 15$, leaf 5 gives $x_{35} = 10$, leaf 3 gives $x_{36} = 5$: the basic solution is feasible.
 - T_2 has no cycle: it is a spanning tree. Leaf by leaf, leaf 1 gives $x_{16} = 30$, leaf 4 gives $x_{24} = 35$, leaf 2 gives $x_{26} = -15$, leaf 5 gives $x_{35} = 10$, leaf 3 gives $x_{36} = 5$: the value -15 is negative, and the basic solution is not feasible.
 - T_3 contains the cells $(2, 5)$, $(3, 5)$, $(3, 6)$, $(2, 6)$, which form the cycle 2, 5, 3, 6, 2 alternating rows and columns: their columns are linearly dependent, and T_3 is not a basis.
2. Let $c'_{ij} = c_{ij} + k$ for every j and $c'_{hj} = c_{hj}$ for $h \neq i$. For every feasible $\mathbf{x}$

$$\sum_h \sum_j c'_{hj} x_{hj} = \sum_h \sum_j c_{hj} x_{hj} + k \sum_j x_{ij} = \sum_h \sum_j c_{hj} x_{hj} + k a_i,$$

because row i of a feasible solution adds up to a_i . All the costs change by the same constant, so the order of the feasible solutions, and the optimal ones, do not change. In the same way, adding k to column j adds $k d_j$ to every cost. In the potential method this is a change of π_i by k (or of π_j by $-k$), up to the normalization, and the reduced costs do not change.

3. The minimum cost rule gives

step	cell	amount	crossed out
1	(1, 6)	20	column 6
2	(3, 5)	10	column 5
3	(1, 4)	10	row 1
4	(2, 4)	20	row 2
5	(3, 4)	5	row 3, column 4

$\bar{x}_{ij}$	4	5	6	a_i
1	10		20	30
2	20			20
3	5	10		15
d_j	35	10	20	

with cost 175. Its potentials and reduced costs are

	4	5	6	π_i
1	$10 \oplus$	10	$20 \ominus$	1
2	20	12	2	3
3	$5 \ominus$	10	$-7 \oplus$	8
π_j	-1	7	0	

- The only negative reduced cost is $\bar{c}_{36} = -7$: the cell (3, 6) enters the basis.
- The cycle is $(3, 6)^+$, $(3, 4)^-$, $(1, 4)^+$, $(1, 6)^-$.
- The cells with sign $-$ have $\bar{x}_{34} = 5$ and $\bar{x}_{16} = 20$: $\theta = 5$, and the cell (3, 4) leaves the basis.
- The cost goes from 175 to $175 + 5 \cdot (-7) = 140$.

The solution is not optimal: one pivot gives the optimum, of cost 140. The minimum cost rule looks at one cell at a time and is not always optimal; this instance is a counterexample.

T_1 is a basis with a feasible basic solution, T_2 a basis with a negative value, T_3 contains a cycle. The minimum cost solution (cost 175) is not optimal: the optimum is 140.

Exercise 7.6 — An assignment of maximum profit

Four salespeople 1, 2, 3, 4 must be assigned to four regions 5, 6, 7, 8, one region each. The expected profits p_{ij} are:

p_{ij}	5	6	7	8
1	9	4	5	1
2	8	5	3	3
3	2	1	1	1
4	2	2	6	8

We look for an assignment of maximum total profit.

Questions

1. Turn the problem into an assignment problem of minimum cost with non-negative costs.
2. Solve it with the Hungarian algorithm.
3. Give the assignment of maximum profit and its profit.

Solution

We use the rules (F1)–(F5) of the box at the beginning of these exercises.

1. The largest profit is $p_{\max} = 9$. With $c_{ij} = 9 - p_{ij}$ every assignment costs $4 \cdot 9$ minus its profit, so minimizing the cost maximizes the profit:

c_{ij}	5	6	7	8
1	0	5	4	8
2	1	4	6	6
3	7	8	8	8
4	7	7	3	1

2. The row minima are 0, 1, 7, 1 and then the column minima 0, 1, 1, 0:

Iteration 1

$\bar{c}_{ij}$	5	6	7	8	π_i
1	0	4	3	8	0
2	0	2	4	5	1
3	0	0	0	1	7
4	6	5	1	0	1
π_j	0	-1	-1	0	

- A maximum set of independent zeros (boxed) has 3 elements: (1, 5), (3, 6), (4, 8).
- The $3 < 4$ lines (blue) are the rows 3 and 4 and the column 5.
- The minimum uncovered entry is $\delta = 2$, in the cell (2, 6) (red). We subtract it from the uncovered entries and add it to the entries covered twice: π_i grows by 2 for the uncovered rows 1 and 2 and π_j grows by 2 for the covered column 5. The dual value grows by $2 \cdot (4 - 3) = 2$.

Iteration 2

$\bar{c}_{ij}$	5	6	7	8	π_i
1	0	2	1	6	2
2	0	0	2	3	3
3	2	0	0	1	7
4	8	5	1	0	1
π_j	2	-1	-1	0	

- A maximum set of independent zeros (boxed) has 4 elements: (1, 5), (2, 6), (3, 7), (4, 8).
- They form an assignment of zero reduced cost: the algorithm stops, and the assignment is optimal.

3. The optimal assignment is $1 \rightarrow 5, 2 \rightarrow 6, 3 \rightarrow 7, 4 \rightarrow 8$, with cost 13 and profit

$$9 + 5 + 1 + 8 = 23 = 4 \cdot 9 - 13.$$

The enumeration of the 24 assignments confirms that it is the unique optimum.

With $c_{ij} = 9 - p_{ij}$ the Hungarian algorithm gives $1 \rightarrow 5, 2 \rightarrow 6, 3 \rightarrow 7, 4 \rightarrow 8$, with profit 23.